STUDY AND ANALYSIS OF DIFFERENT CHANNEL ENCODING AND DECODING TECHNIQUES

Thesis submitted in the partial fulfilment of the requirements of

Bachelor of Technology

In

ELECTRONICS AND COMMUNICATION ENGINEERING

BY

RONIT KUMAR

111EC0175

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

ROURKELA-769008

ODISHA,INDIA

MAY 2015

STUDY AND ANALYSIS OF DIFFERENT CHANNEL ENCODING AND DECODING TECHNIQUES

DISSERTATION SUBMITTED IN MAY 2015

TO THE DEPARTMENT OF

ELECTRONICS AND COMMUNICATION ENGINEERING

Of

National Institute of Technology,Rourkela

In partial fulfilment for the requirements of

Bachelor Of Technology

By

RONIT KUMAR(111EC0175)

Under the supervision of

Prof.Upendra Kumar Sahoo

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

ROURKELA-769008

ODISHA,INDIA

MAY 2015

CERTIFICATE

This is to certify that the work on the thesis entitled **Study and analysis of different channel encoding and decoding techniques** by **Ronit Kumar** is a record of original research work carried out under my supervision and guidance for the partial fulfilment of the requirements for the degree of **Bachelor in Technology** in the department of **Electronics and Communication Engineering** in **National Institute of Technology,Rourkela.**

Place:NIT ROURKELA

Date:MAY 2015

Prof.Upendra Kumar Sahoo

Professor,ECE Department,

NIT ROURKELA,ODISHA

ACKNOWLEDGEMENT

First and foremost I would like to express my sincere gratitude towards my project supervisor and guide Prof. Upendra Kumar Sahoo for his advice during my project work.H e has encouraged me all through out the year which helped me in achieving my goal.His observations helped me to establish the overall direction of the research and to move forward with investigation in depth.He has helped me greatly and been a source of knowledge.

I would like to extend my thanks to our HOD,Prof.Kamalakanta Mahapatra and to all the professors of the department for their support.

I would like to thank administrative and technical staff members of the Department who have been kind enough to advise and help me in their respective role.

Last but not the least,I would like to acknowledge the love,support and motivation I received from my parents and therefore I dedicate this thesis to my family.

RONIT KUMAR

111EC0175

ABSTRACT

Maintenance of the quality of data is the most important thing in communication. There are different components that influence the nature of information when it is transferred over a communication channel like noise, fading etc. To conquer these impacts channel coding schemes are introduced. In a digital communication system we have to transmit more data via a noisy channel which results in vast no. of errors. The answer for this issue is channel encoding and decoding techniques which helps in error control and in increasing the effectiveness and performance of communication systems.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Objective of the thesis:

Analysing and MATLAB implementation of the Convolutional codes and its encoding and decoding techniques

Analysing and MATLAB implementation of the Turbo codes, its encoder and decoder structures and their performance through BER charts

Analysing and MATLAB implementation of the performance of Turbo code through EXIT Charts

Analysing and MATLAB implementation of the performance of Repeat Accumulate codes through EXIT Charts and BER charts

1.2 Organisation of the thesis

Chapter 2-This chapter analyses the advantages of convolutional codes over linear block coding techniques. It also describes its encoding and decoding procedures

Chapter 3- This chapter analyses the advantages of Turbo codes over convolutional codes .It also describes its encoding and decoding procedures and their structures.

Chapter 4- This chapter analyses the advantages of analysis of Turbo codes through EXIT Charts as compared to the BER curves.

Chapter 5- This chapter analyses the advantages of repeat accumulate codes over turbo codes. They are also analysed through EXIT charts.

CHAPTER 2

CONVOLUTIONAL ENCODING AND DECODING

2.1 BASICS

Convolution codes don't process the data bits block wise rather they process the consecutive bits and use the convolution property of the polynomials to generate the code words. This is in contrast to the block codes which process the data in a blocked manner.

The contrast between block codes and convolutional codes is the encoding guideline. It is a sort of error remedying codes.A polynomial function helps in sliding the stream of information bits. This is equivalent to the convolution operation which justifies the name of these codes.

A convolutional encoder encodes L data bits to M > L code bits in every time step. The encoding technique is not memoryless, as the code bits rely on upon the data bits encoded at past time steps. This is another enormous distinction from block codes, as the block codes are memoryless. The reality that the convolutional codes have memory permits them to work well, when L and M are really little. The block codes need to have long block lengths , in light of the fact that they are memoryless and their execution enhances with block length.

2.2 Convolutional Encoding

2.2.1: System model

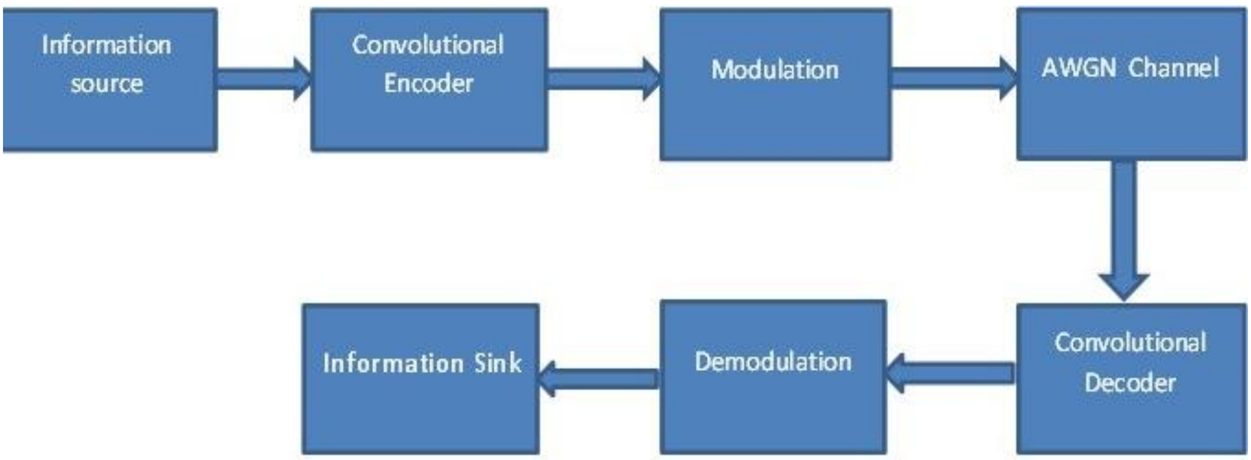


Figure 1: Block diagram of convolutional encoder

- A succession of message bits is utilized as information source Convolutional coding is connected on the binary information . After the completion of encoding of data, modulation is performed.

- AWGN noise is included when this coded information is gone through the channel. The noise present in the channel changes a portion of the information bits.

- Demodulation of the got sequence is performed at the recipient end. After that convolutional decoding is performed which helps us to recover the unique transmitted message.

2.2.2. ENCODING PROCEDURE

- A bit is moved into the furthest left stage at every information and the bits beforehand existing in the movement registers are moved one position to right. In the wake of applying the modulo-2 operation relating yields are gotten. This procedure of proceeds until the landing of information at the info of encoder. The decision of association between the movement registers and adders depicts the attributes of code. By fluctuating the associations, attributes of the code can be changed.

- To portray an encoder, set of “n” association vectors are needed.. These associations depict which shift register is joined with m adders. A “1” in the position shows that, shift register is joined with the adder and a “0” in given position will demonstrate that not a solitary association exits between the stage and the adder. In the figure the upper and lower connection vectors are h1=[1 1 1] and h2=[1 0 1].

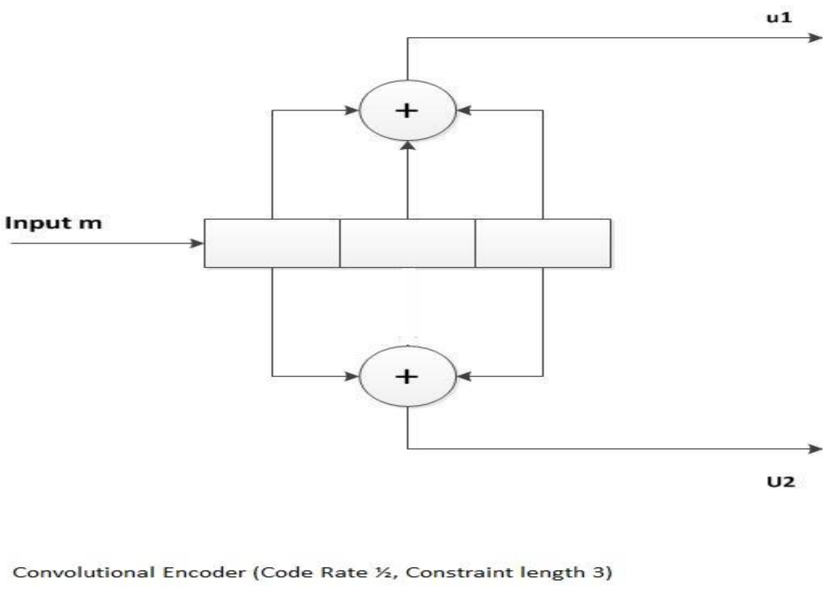


Fig 2:Convolutional Encoder

2.2.3 Impulse Response of the Encoder

An encoder can be surveyed on the premise of its impulse response. When a high bit is passed through the encoder we get its impulse response. Consider the contents of the register as a "1" is passed through it:

MEMORY	OUTPUT(U1)	OUTPUT(U2)
000	0	0
100	1	1
010	1	0

TABLE 1:IMPULSE RESPONSE

In the above case the input sequence is :0 1 0 and the output sequence is
00 11 10.The symbol "11" is the impulse response.

2.2.4 Convolutional Encoder using time domain approach

The time domain conduct of a binary convolution encoder having code rate 1/N may be characterized as far as set of N impulse responses.. Lets consider a basic encoder having code rate ½.We need two impulse responses to describe its conduct in the time space. Let the sequence [g01 , g11 ,… … , gx1]{Here ,x speaks to the quantity of points from where bits are extricated for modulo-2- adders}denotes the impulse response for way p1 and the grouping [g02 , g12 ,… … , gx2]denotes the impulse response of way p2. The characterized impulse response is known as the generator sequence of the code. (d0, d1,d2,…..) denote the message sequence that enter the encoder one bit at a time (starting with d0).The encoder convolves the message sequence with the impulse response of path p1 and p2 respectively and then generates two output sequences described as Ci^1 and Ci^2.The output sequence of path p1 is the convolutional sum $Ci^{(1)}=\Sigma^{x}_{i=0}\ gi^{(1)}\ d_{i-1}\quad {\scriptstyle i=0,1,2\ldots}$

2.2.5 Tree diagram

The tree diagram helps in understanding the coding procedure of these type of codes. The encoded group of bits can be derived from the tree diagram. It makes the encoding procedure more detailed and easy to comprehend.

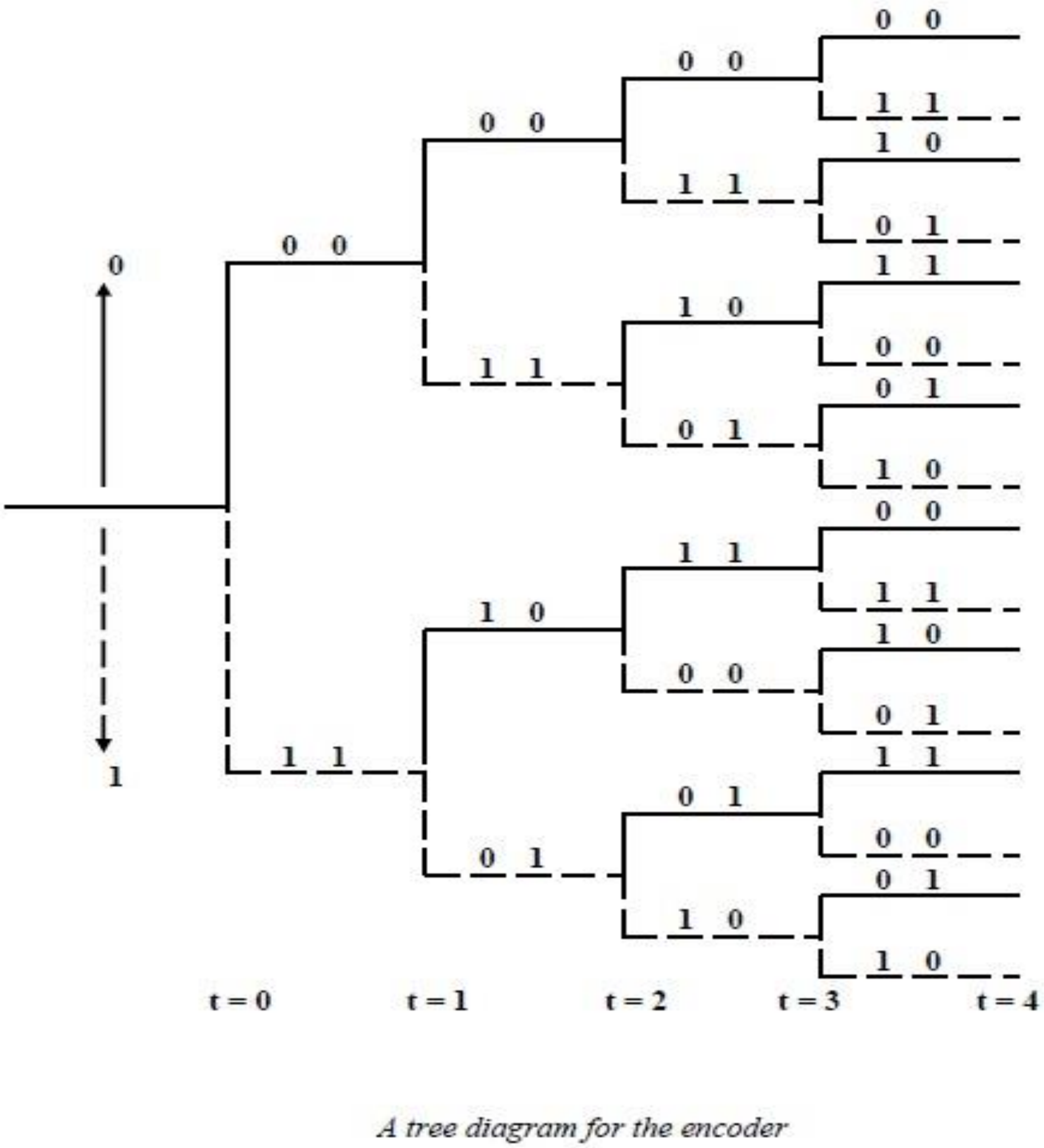


Fig 3-Code tree

Each branch of the tree represents an input symbol and the corresponding pair of binary symbols(output) is shown on the branch. An input 0 indicates the upper and 1 represents the lower branch.

2.2.6 Trellis diagram

The trellis diagram of a convolutional code is acquired from its state diagram. All state moves at every time step are expressly demonstrated in the graph to hold the time dimension, as is available in the relating tree diagram. Normally, supporting portrayals on state moves, relating info,input and output bits and so on are named in the trellis diagram. It is fascinating to note that the trellis chart, which portrays the operation of the encoder, is extremely helpful for portraying the conduct of the relating decoder, particularly when the celebrated 'Viterbi Algorithm (VA)' is taken after.

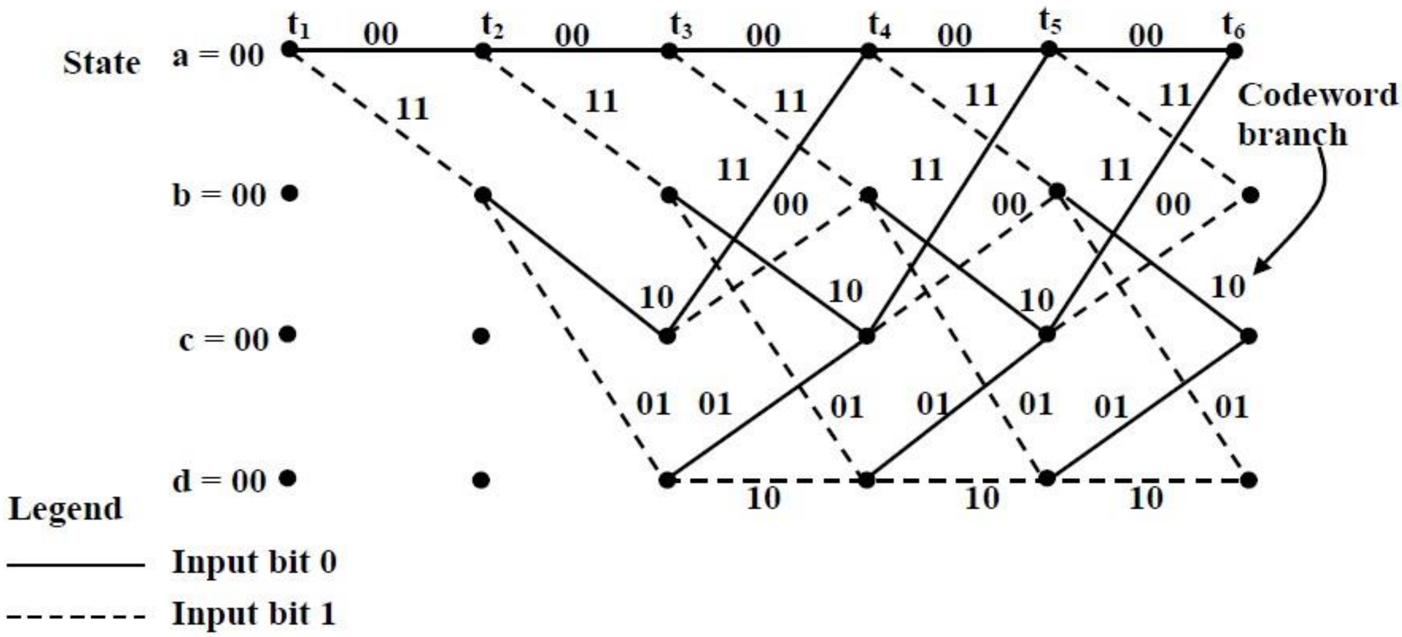


Fig 4:Trellis diagram of the encoder

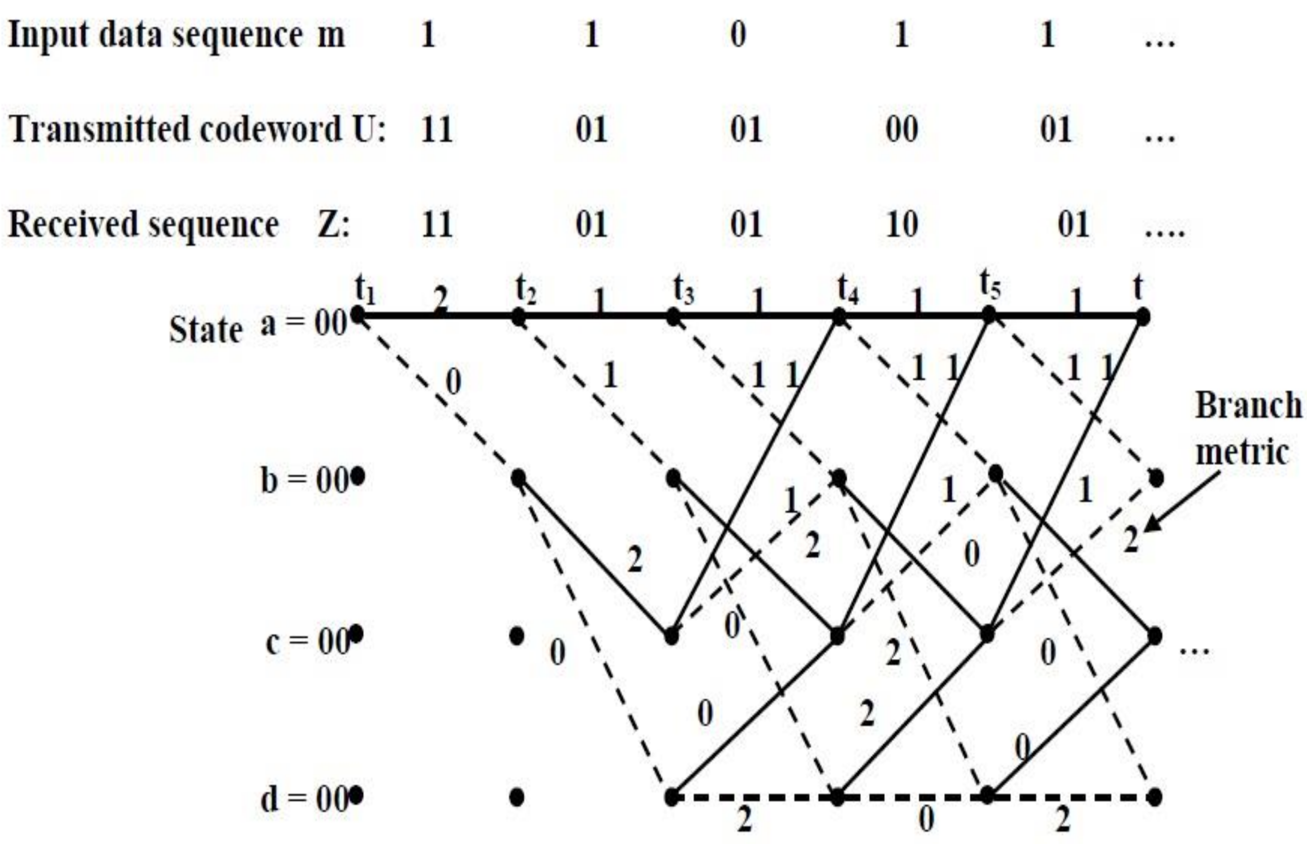


Fig 5:Trellis diagram of the decoder

2.2.7 State Diagram

Substance of the furthest right (K-1) shift register stages characterize the states of the encoder. Along these lines, the encoder in Fig. has four states. The move of an encoder starting with one state then onto the next, as created by data bits. Fig.4 demonstrates the state diagram of the encoder in Fig.2. Another information bit causes a move starting with one state then onto the next. The way data between the states, signified as b/c1 c2 , speaks to information data bit "b" and the corresponding output bits (c1c2).

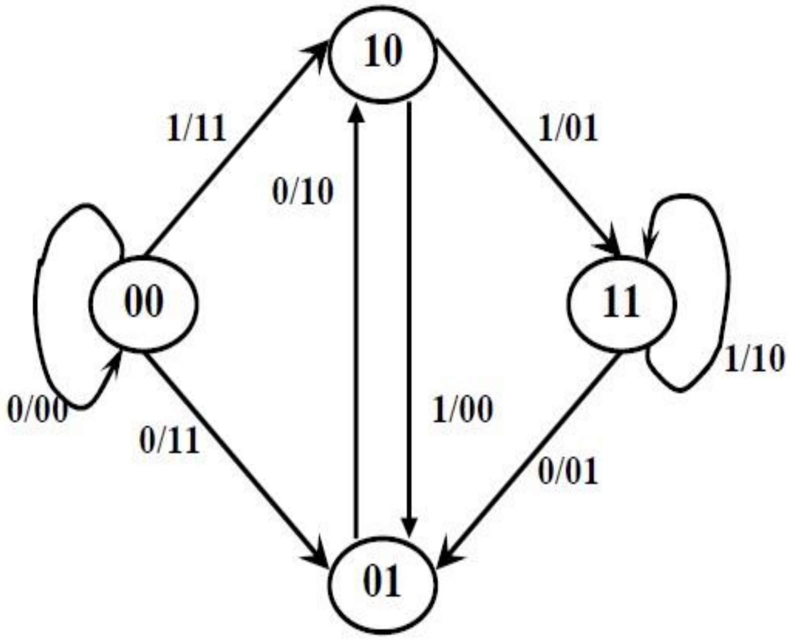


Fig 6:State diagram of the encoder in fig 2

2.3 Convolutional decoding

2.3.1 Maximum likelihood decoding

- It helps us to find the shortest path in the trellis diagram.
- It is based on the calculation of hamming distances.
- Decoding probability is:

$$p(y,x) = \prod_{j=0}^{\infty} p(y_j | x_j)$$

2.3.2 Viterbi Algorithm

- ML algorithm is too complex to search all available paths because of end to end calculation.

- Viterbi algorithm performs ML decoding by lessening its complexity dispensing with the most unlikely trellis paths at every transmission stage .It additionally diminishes decoding complexity with ahead of schedule dismissal of unlike paths.

- Viterbi algorithm gets its efficiency via concentrating on survival paths of the trellis.

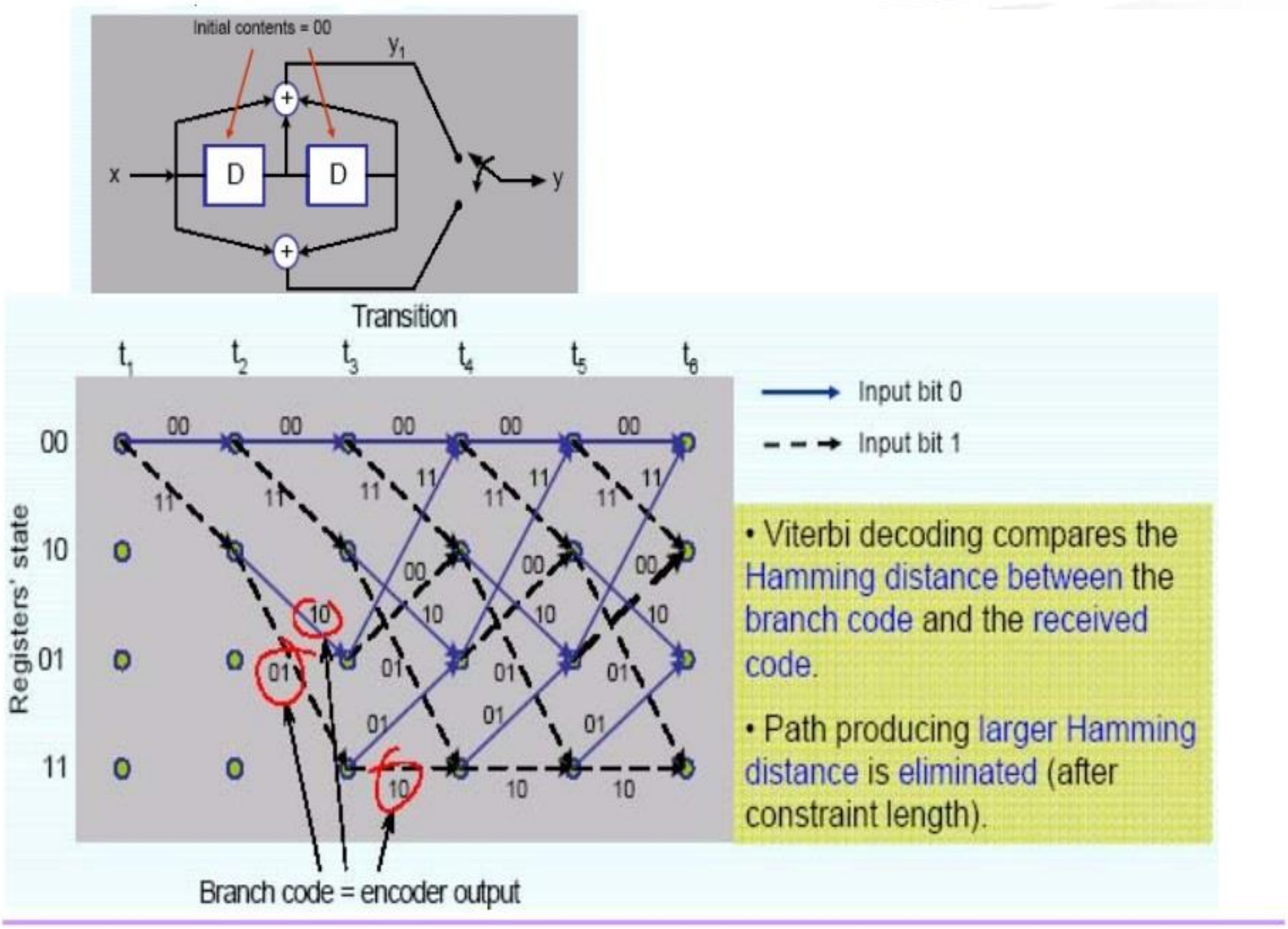


Fig 7:Viterbi decoding

EXAMPLE OF VITERBI DECODING

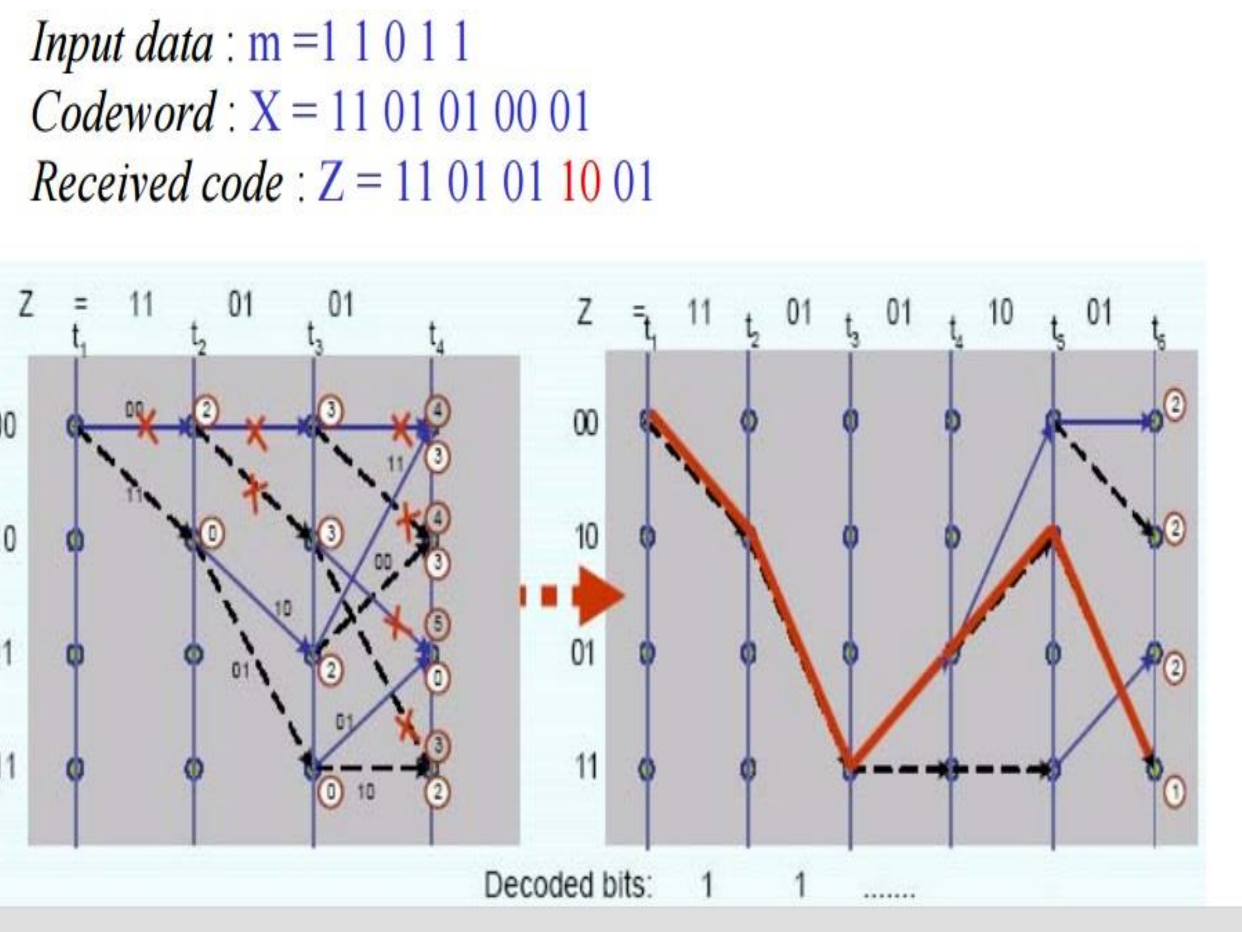


Fig 8:Example of Viterbi decoding

2.3.3 Simulation result

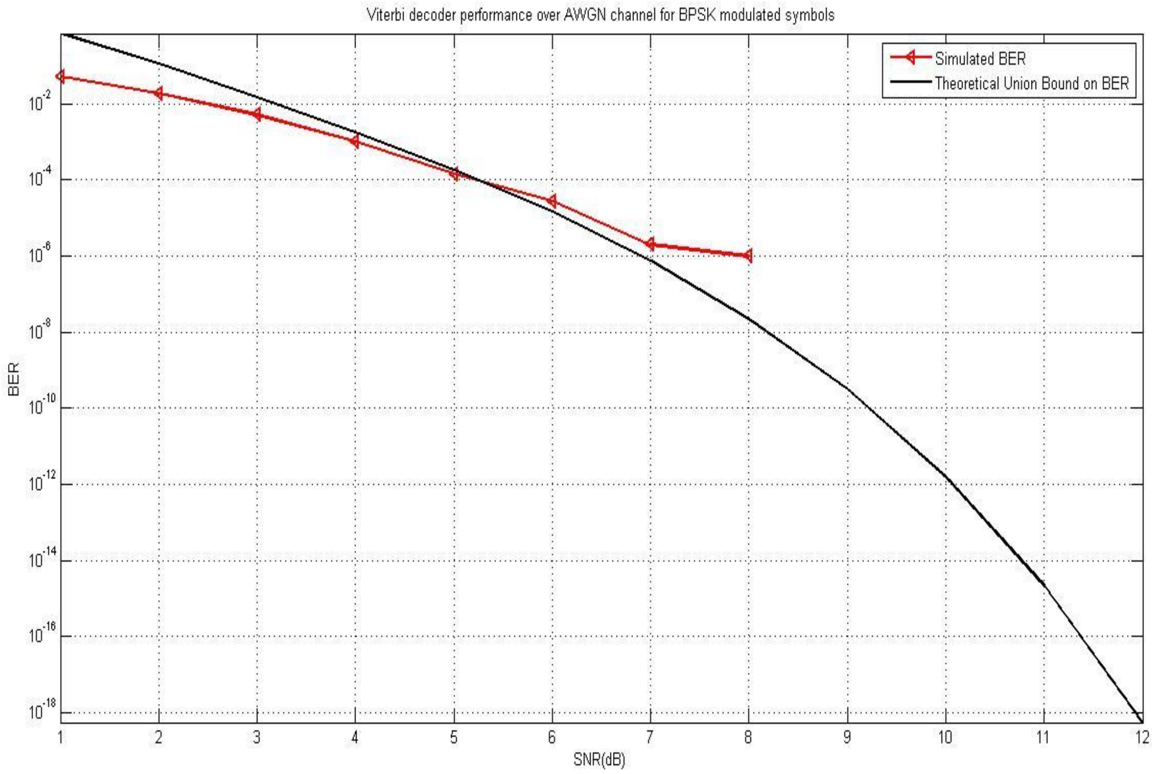


Fig 9 : BER Simulation of turbo decoder performance

2.3.4 CONCLUSION

This chapter analyses the convolutional coding and decoding methods and proves that it is better than the block codes.

CHAPTER 3

TURBO EQUALISATION

3.1 ABSTRACT

Turbo equalisation is a characteristic augmentation to the advancement of iterative algorithms for decoding. If the data bits are protected by FEC and has to be sent through a ISI channel, then this method is really helpful.

3.2 BASICS

- Below given is the transmitter of a communicaton system. The parts of this system are very important if we want to utilize turbo equalization in the receiving side.

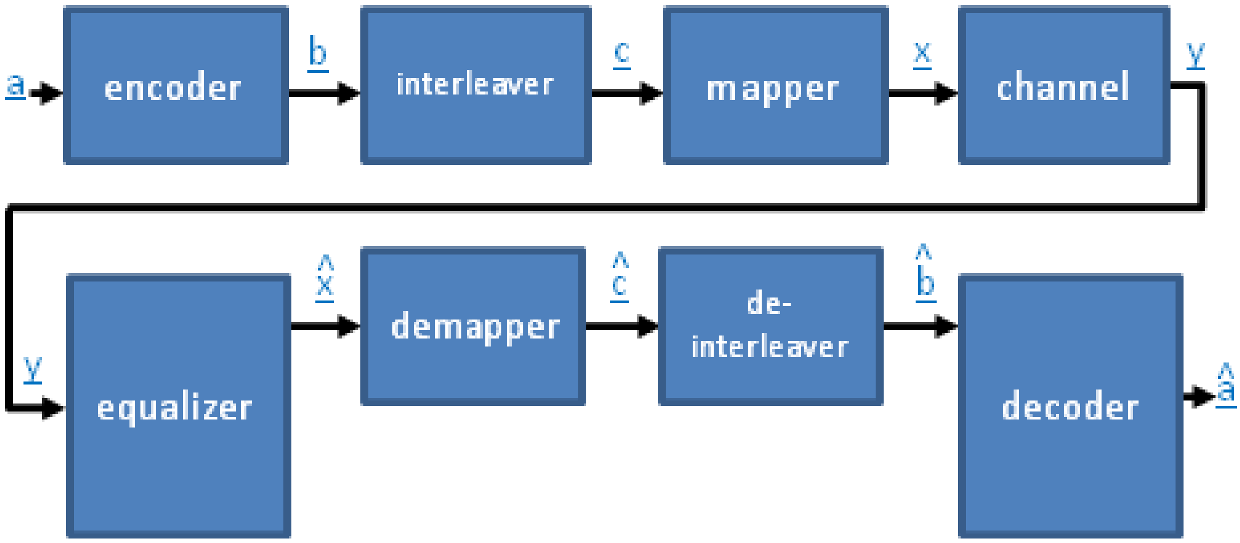


FIG 10:DIAGRAM OF A COMMUNICATION SYSTEM

The basic elements in the transmitter are-

1.Encoder
2.Interleaver
3.Mapper
4.Channel

ENCODER- It takes a binary data sequence and produces the output which contains redundant information which protects the useful data from error during transmission.
Convolutional codes are used to produce the redundant information.

INTERLEAVER- The aim of FEC codes is to shield the data from single bit error or short burst error that is introduced due to noise in the channel.. Because of its shuffling effect it decorrelates the error events introduced or which can't be resolved by the equalizer between neighbouring symbols. Using only a convolutional decoder makes it difficult to mitigate the error bursts.

MAPPER- It maps the binary bits into channel symbols . In this way binary data are converted into electrical signal which is then mapped into the channel. The conversion to channel symbols makes it suitable for modulation.

Receiver- Receiver optimally estimates the data that was transmitted.

- Receiver estimates the data such that there is minimum bit error rate. The complexity of receiver increases because of so many factors.
- The complexity increases exponentially with the length of data

In this methodology for expanding the performance, equalizer is utilized for minimizing the mean square error and symbol error rate by amplifying likelihood of observation in the channel.

3.3 TURBO EQUALIZER

- Equalization mitigates the effects of an ISI channel or detection.

- The complexity of Viterbi and BCJR is high for certain channels.
- The equalization is dependent upon the resulting information from the decoder.
- It has an additional loop of feedback as compared to the other equalizers.

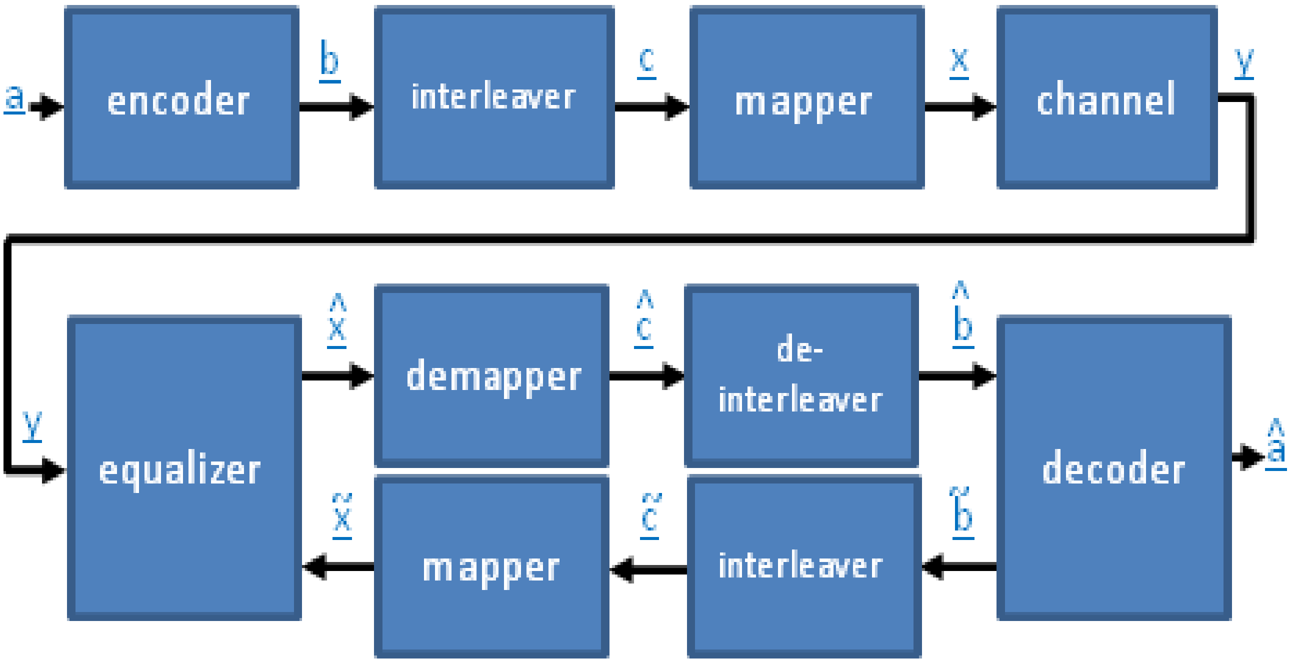


Fig 11:Block diagram of turbo equalizer

- It has application in TCM and CDMA.

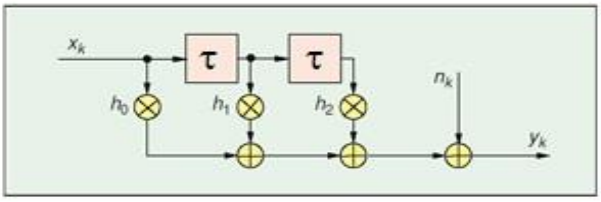
Turbo equalization helps in the reduction of the count of states.

3.3.1 Used channel model

In the following, we are using an AWGN channel with known channel impulse response (CIR). The received signal is given by

$$y_k = \sum_{l=0}^{L} (h_l \cdot x_{k-l}) + n_k \quad k=1,2,\ldots,N$$

channel coefficient   sent signal   noise

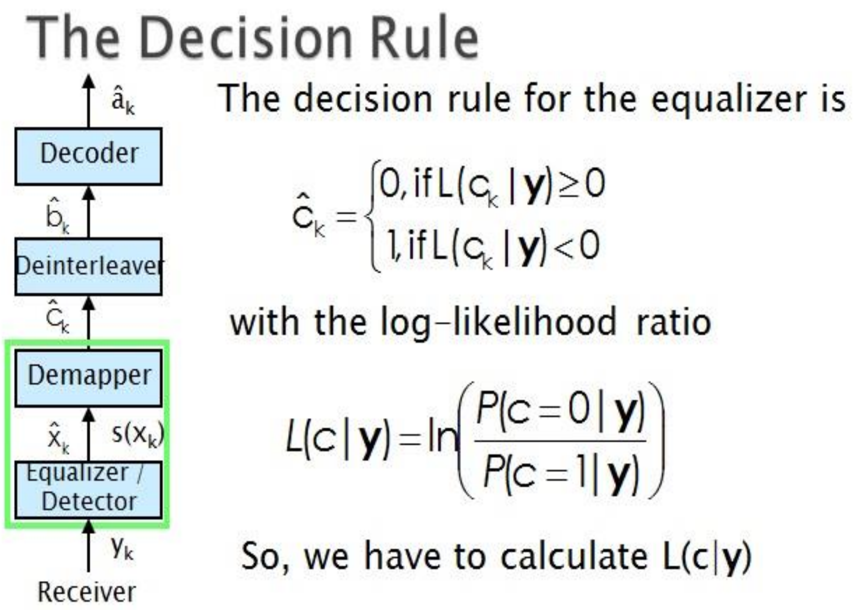


In matrix form: $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} h_0 & 0 & 0 & 0 & \cdots & 0 \\ h_1 & h_0 & 0 & 0 & \cdots & 0 \\ h_2 & h_1 & h_0 & 0 & \cdots & 0 \\ 0 & h_2 & h_1 & h_0 & & 0 \\ & \ddots & \ddots & \ddots & \ddots & \\ 0 & 0 & \cdots & h_2 & h_1 & h_0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_N \end{bmatrix}$$

The noise is Gaussian $p(n) = \frac{e^{\frac{-n}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}}$

3.3.2 Detection



- LLR's make or work easier as compared to probabilities.
- The extrinsic information is related to $L(c_n)$ according to the equation
- $L(c_n|y)=L_{ext}(c_n|y)+L(c_n)$
- $L_{ext}(c_n|y)$ represents the extrinsic information about a_n contained in y.
- The above quantities are very vital for the process of detection.

Presented below is block diagram of the turbo equalisation showing the intrinsic, extrinsic, a priori and a posteriori probabilities.

.

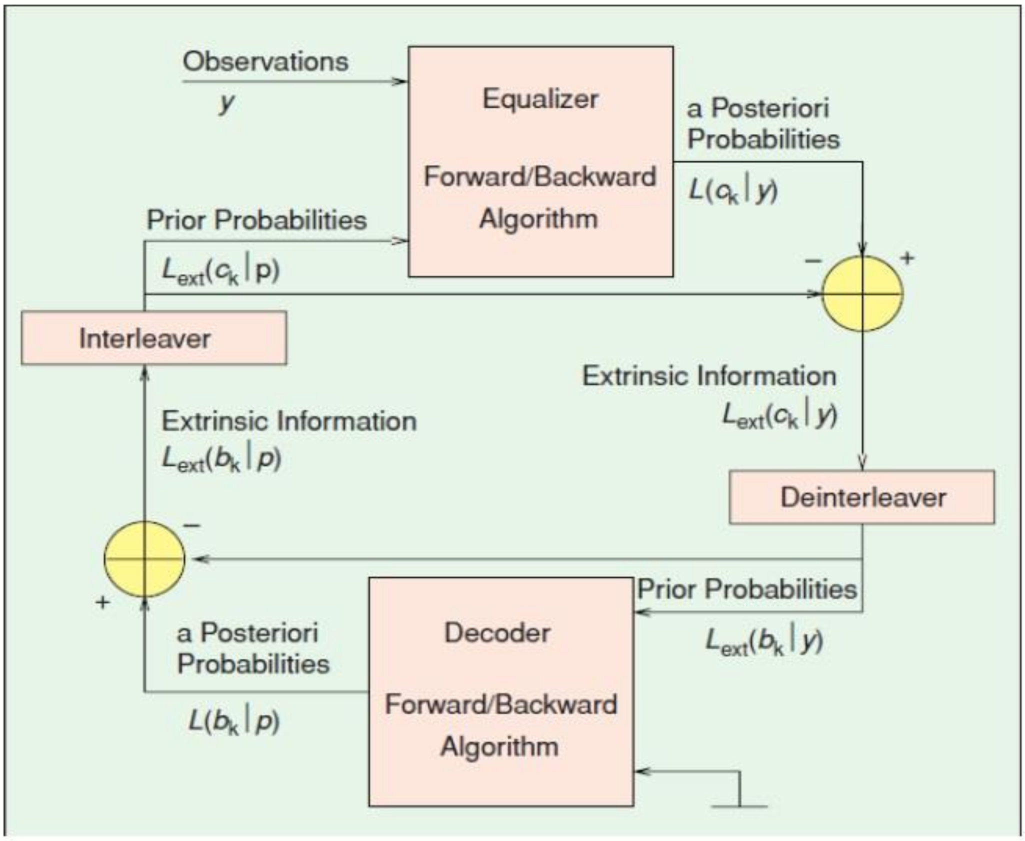


Fig 12:Block diagram of turbo equalisation

3.3.3 Turbo equalization algorithm

Input: Channel coefficients h_l for $l = 0, 1, \ldots, L$.
Observation sequence y.
A sequence of LLRs $L_{ext}(c|p)$ initialized to 0.
A predetermined number of iterations ℓ.

Recursively compute for ℓ iterations:

$$L(c|y) = \text{Forward/Backward}\left(L_{ext}(c|p)\right)$$
$$L_{ext}(c|y) = L(c|y) - L_{ext}(c|p)$$
$$L(b|p) = \text{Forward/Backward}\left(L_{ext}(b|y)\right)$$
$$L_{ext}(b|p) = L(b|p) - L_{ext}(b|y)$$

Output: Compute data bit estimates $\hat{a}_k$ from $L(a_k|y)$.

3.4 Turbo encoder

Turbo encoder consists of two RSC encoders as shown in the fig. The data bits are taken by the first encoder and it generates the parity bits. The interleaver shuffles the data bits. The second encoder uses the interleaved information and it generates its parity bits.

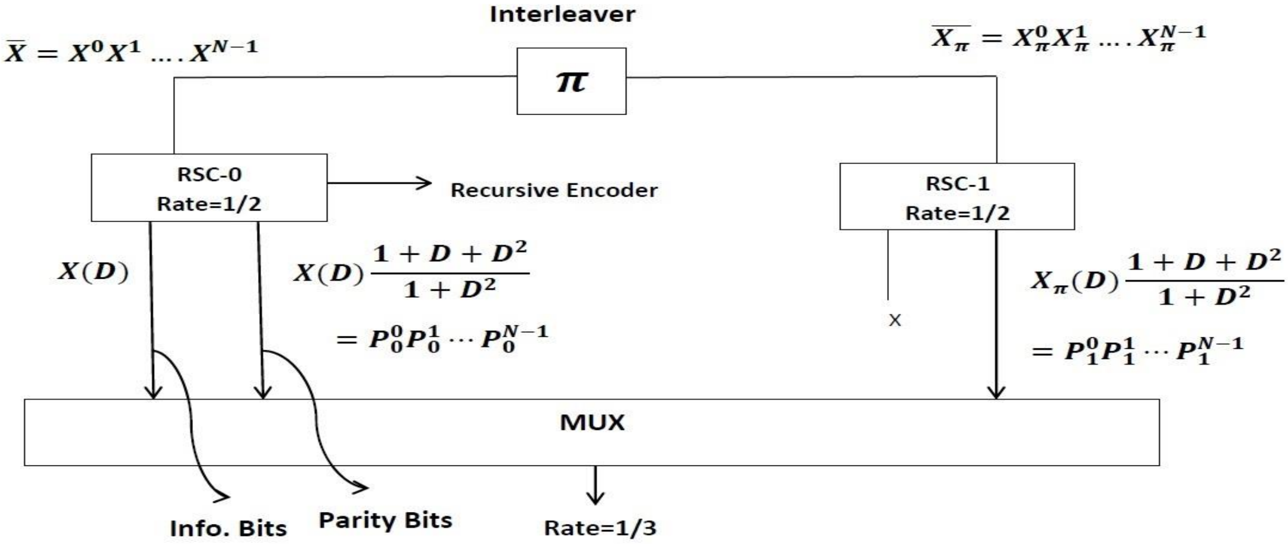


Fig 13: Structure of the turbo encoder

The process of the encoding is as described below:

$$G(D) = \left[1 \quad \frac{1+D+D^2}{1+D^2}\right]$$

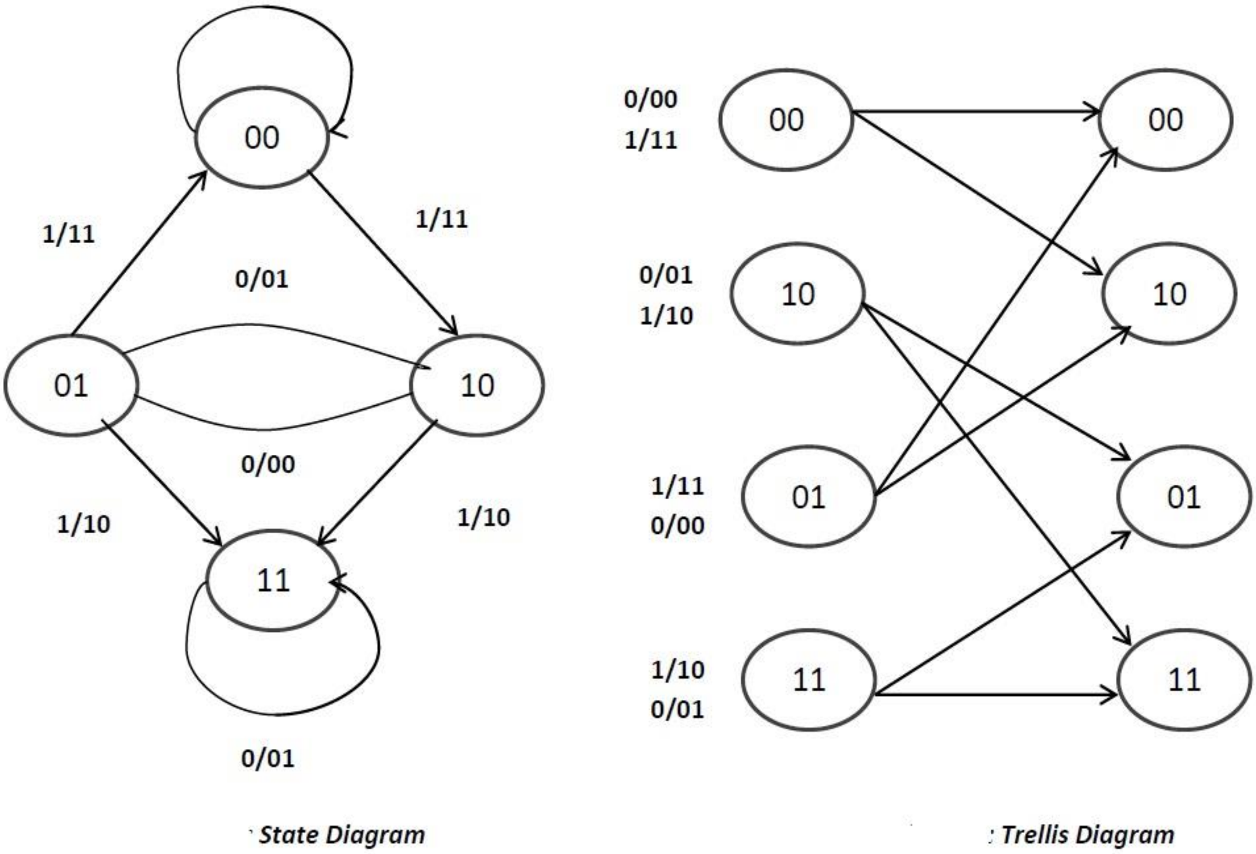


Fig 14:State and trellis diagram

3.5 Turbo decoder

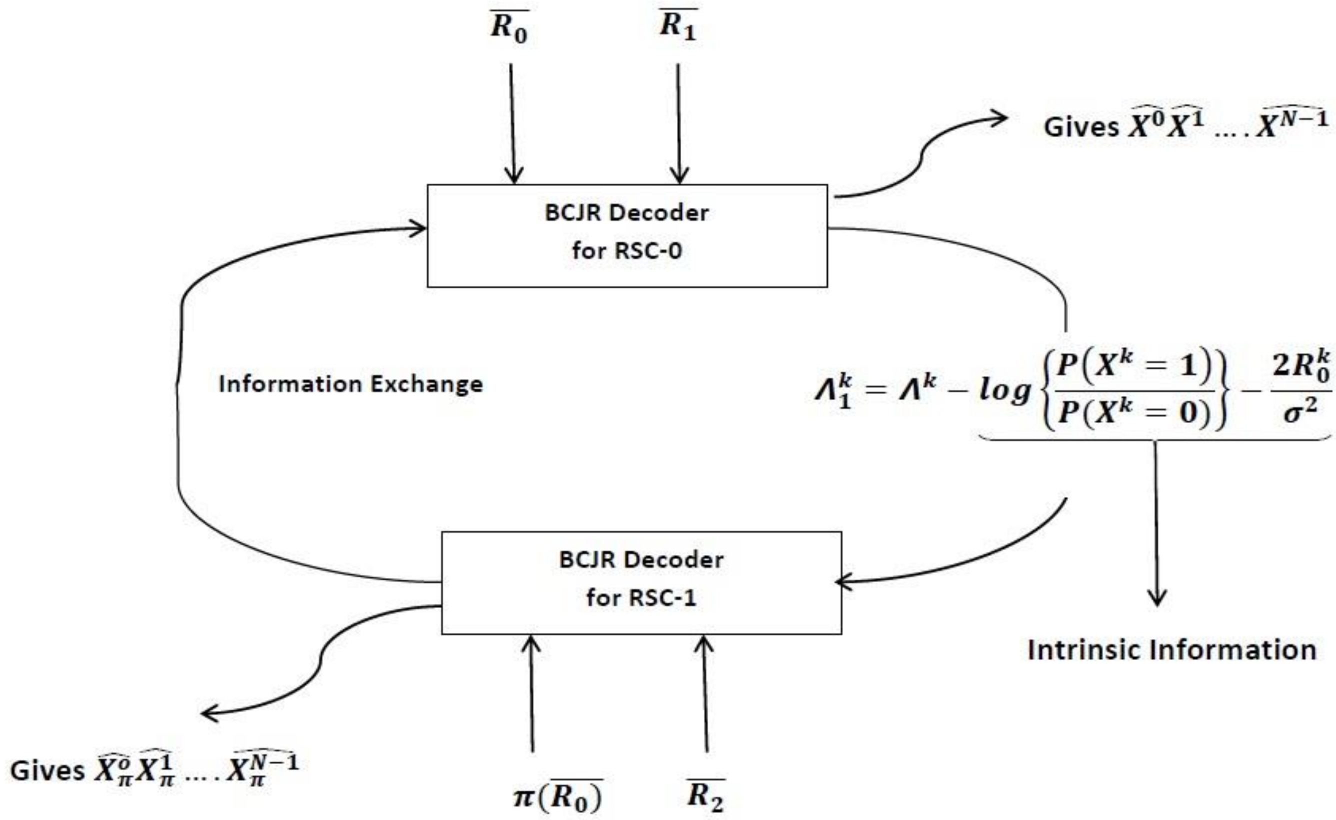


Fig 15:Turbo decoder architecture

Above shown is the architecture of two BCJR decoders i.e decoder 0 and decoder 1.The input to the first decoder are R0 and R1.The second decoder is given ∏(R0) and R2 as inputs.

The Log A posteriori probability ratio is described below:

$$\Lambda^k = log\left\{\frac{P(X^k = 1|\overline{R})}{P(X^k = 0|\overline{R}}\right\}$$

For k=0,1,....N-1.

where $\overline{R}$ is the input to any of the decoder. Consider BCJR-0, then $\overline{R} = \overline{R_0}\ \overline{R_1}$
It can be shown that

$$P(X^k = i|\overline{R}) = \frac{1}{P(R_0^{N-1})}\sum_{m=0}^{M-1}\sum_{m'=0}^{M-1}\alpha_k(m')\gamma_k^i(m',m)\beta_k(m)$$

Where i=0 or 1 (input), m and m' are stages (S_k), M is total number of stages. Stage m=0, 1, 2 and 3 respectively represents stage 00, 01 10 and 11.

The description of terms inside summation is given as follows:

$\alpha_k(m')$ is forward state metric

$$\alpha_k(m') = P(S_k = m'|\overline{R_0^{k-1}})$$

Solving $\alpha_k(m')$ gives

$$\alpha_k(m') = \sum_{m''=0}^{M-1}\sum_{i=0}^{1}\gamma_{k-1}^i(m'',m')\alpha_{k-1}(m'')$$

$\beta_k(m)$ is backward state metric

$$\beta_k(m) = P(\overline{R_{K+1}^{N-1}}|S_{k+1} = m)$$

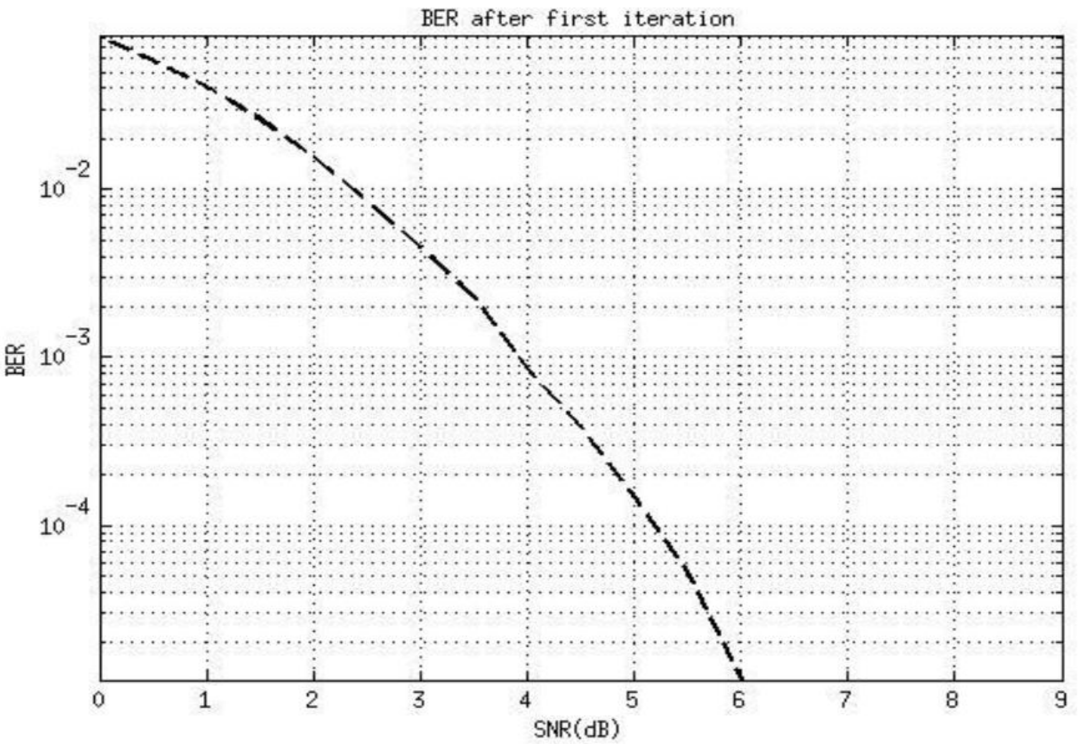
Solving $\beta_k(m')$ gives

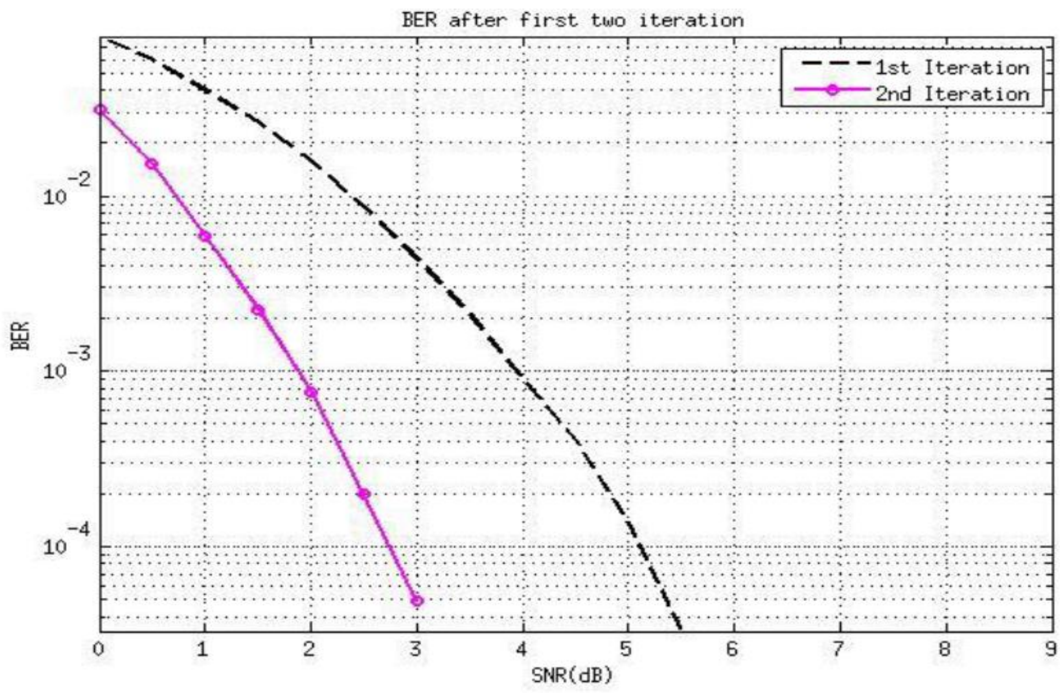
$$\beta_k(m) = \sum_{m'=0}^{M-1}\sum_{i=0}^{1}\gamma_{k+1}^i(m',m)\beta_{k+1}(m')$$

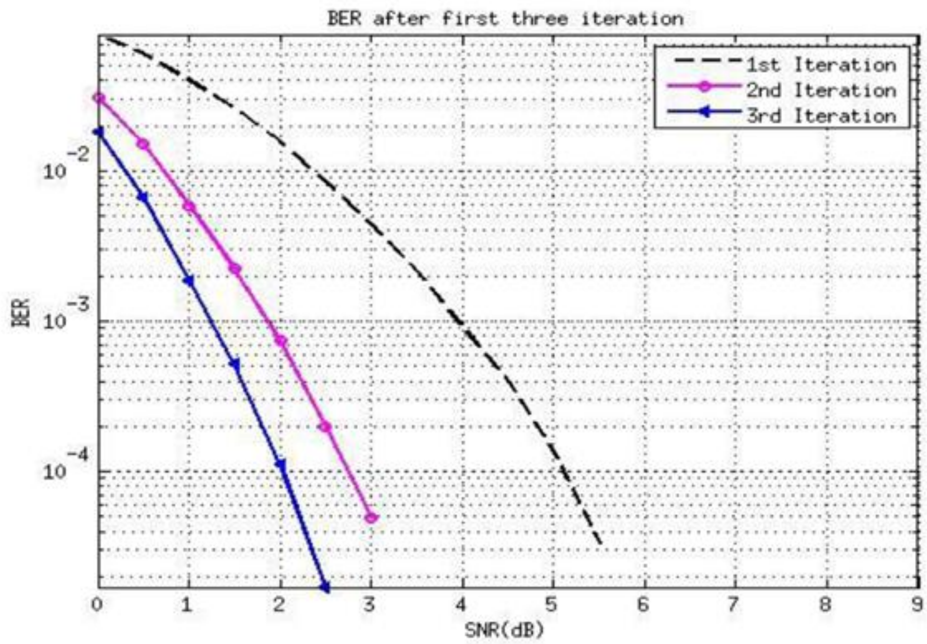
$\gamma_k^i(m',m)$ is transition probability from state m' to m at stage k for input i

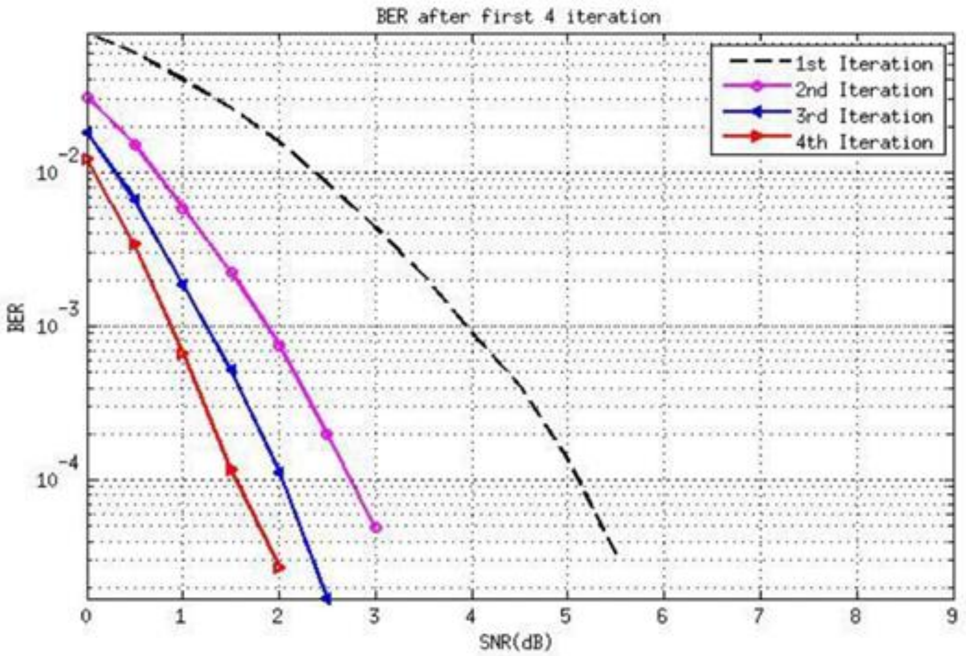
$$\gamma_k^i(m',m) = \underbrace{P(X^k = i)}_{\text{Apriori Probability at time k for input i}}\ \underbrace{P(S_{k+1} = m|S_k = m', X^k = i)}_{\text{Obtained from trellis}}\ \underbrace{P(\overline{R^k}|S_{k+1} = m', S_k = m, X^k = i)}_{\text{Obtained from likelihood}}$$

3.6 Simulation results of Turbo decoder




BER after first two iteration
1st Iteration
2nd Iteration
BER
SNR(dB)


BER after first three iteration
1st Iteration
2nd Iteration
3rd Iteration
BER
SNR(dB)



3.7 CONCLUSION

This chapter summarizes the turbo codes,its advantages over convolutional codes.It also presents and explains the operation of turbo encoders and decoders.The performance of turbo codes is analysed through BER charts.

CHAPTER 4

EXIT CHARTS FOR TURBO CODES

4.1 MOTIVATION AND INTRODUCTION

The problem with Bit Error Rate Chart when iteratively decoding is that the simulation of small BER values take a lot of time and do not provide much details of performance. But other performance analysis techniques can evaluate the ber curves much quickly. Partially-analytical apparatuses contrived for exploring the convergence behaviour of iteratively decoded frameworks were proposed. The exchange of extrinsic information between the decoders is shown by the mutual information between the transmitted bits and the decoded bits at the receiver. The trading of extrinsic data between the constituent decoders may be pictured by Extrinsic Information Transfer (EXIT) diagrams .The analysis of EXIT charts was improved for the situation when the PDF of the transmitted message is symmetric. The exchange of extrinsic information is visualized as a decoding trajectory in EXIT Charts. It shows good performance in low E_b/N_0 & turbo cliff regions.It helps in designing better. codes.

4.2 Construction of Exit charts

The ingredients which we need for the construction of EXIT charts are:

1. Mutual information

2.Mutual information transfer characteristics of iterative decoders

3.Combination of transfer characteristics

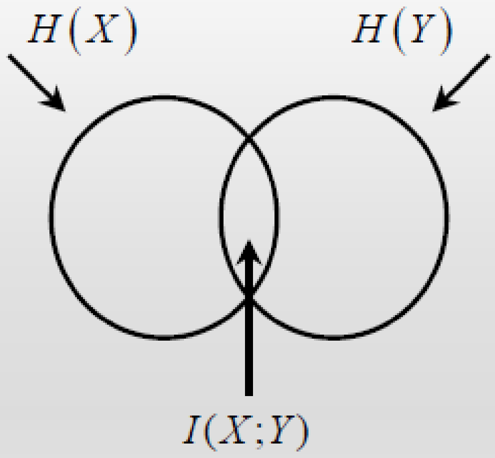
Mutual information

$$I(X;Y)=H(Y)-H(Y|X)$$

$$I(X;Y)=\iint f(x,y)\log\frac{f(x,y)}{f(x)f(y)}dxdy$$

where

$$H(Y|X)=\iint f(x,y)\log\frac{1}{f(y|x)}dxdy$$



Channel Capacity

$$C=\max_{p_x} I(X;Y)$$

4.2.1 Benefits of mutual information in EXIT charts

- We get an additional interpretation based on information theory
- Due to logarithmic scaling we can accommodate a large range of values.
- The shape of the curves is robust and provides a lot of information.
- They have an additional area property

4.2.2 **Iterative decoder**

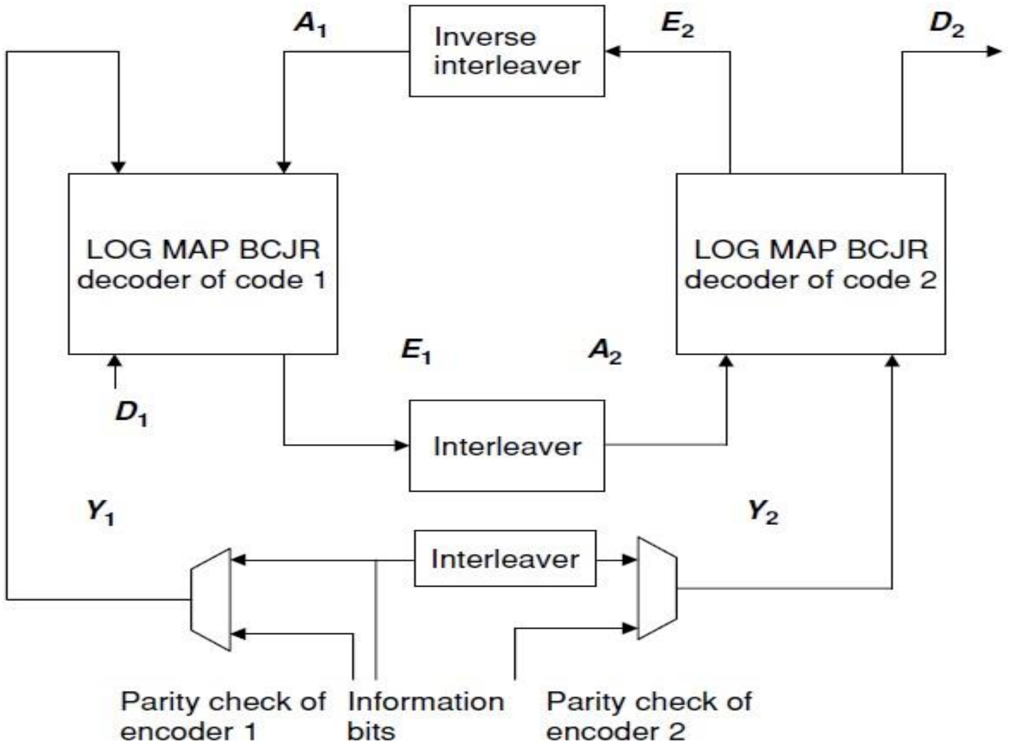


Fig 16: Block diagram of the Log Map BCJR decoder

Equations used:

The extrinsic information vector is:

$\boldsymbol{E}1 = \boldsymbol{D}1 - \boldsymbol{A}1 - \boldsymbol{Y}1$ with components

$Ei1 = Le1(bi) = L1(bi/\boldsymbol{Y}1) - L1(bi) - Lc\, yi1 = Di1 - Ai1 - Yi1$ is generated by the first decoder.

- $\boldsymbol{E}2 = \boldsymbol{D}2 - \boldsymbol{A}2 - \boldsymbol{Y}2$ whose components are

$Ei2 = Le2(bi) = L2(bi/\boldsymbol{Y}2) - L2(bi) - Lc\, yi2 = Di2 - Ai2 - Yi2$

- The first decoder accepts the intrinsic data and outputs soft values S1.
- The estimates are used by the first decoder to manufacture extrinsic information.
- It is interleaved and is equivalent to the a priori info for the second decoder
- The second decoder produces its extrinsic information which becomes the a priori information of the first decoder.

4.2.3 Regions of the BER Curve

BER performance curve of a turbo code basically has 3 regions:

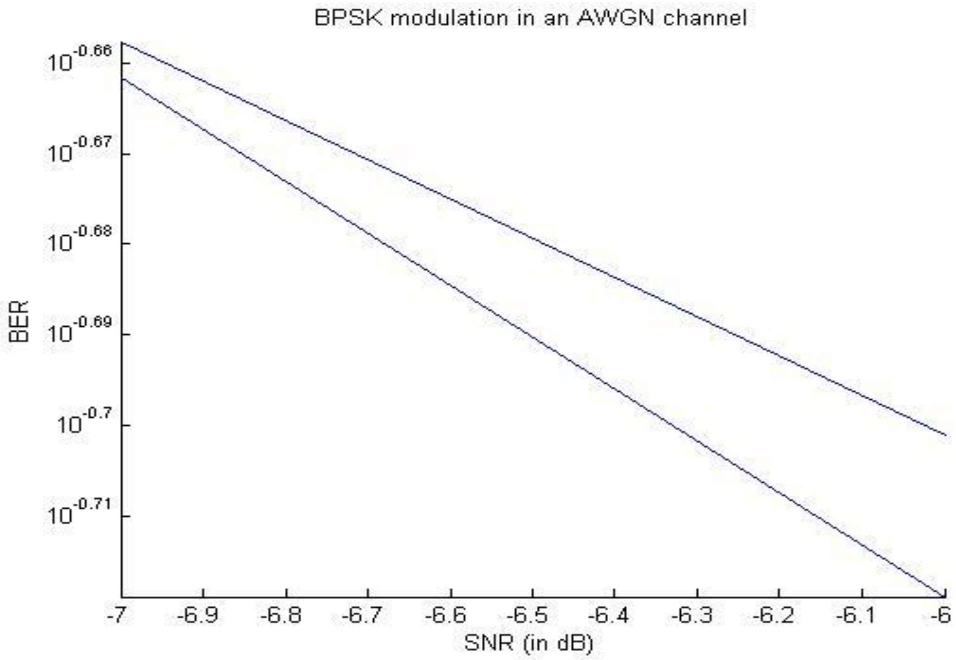
- In the 1st region (low values of SNR) the performance of the iterative decoding is less as compared to the uncoded one which doesn't even improve by taking a large no. of iterations.
- In the 2nd region at (middle values of SNR), iterative decoding has an excellent performance. The performance increases in this so called waterfall region.
- In the 3rd region (high values of SNR),the rate of performance increment decreases with SNR.

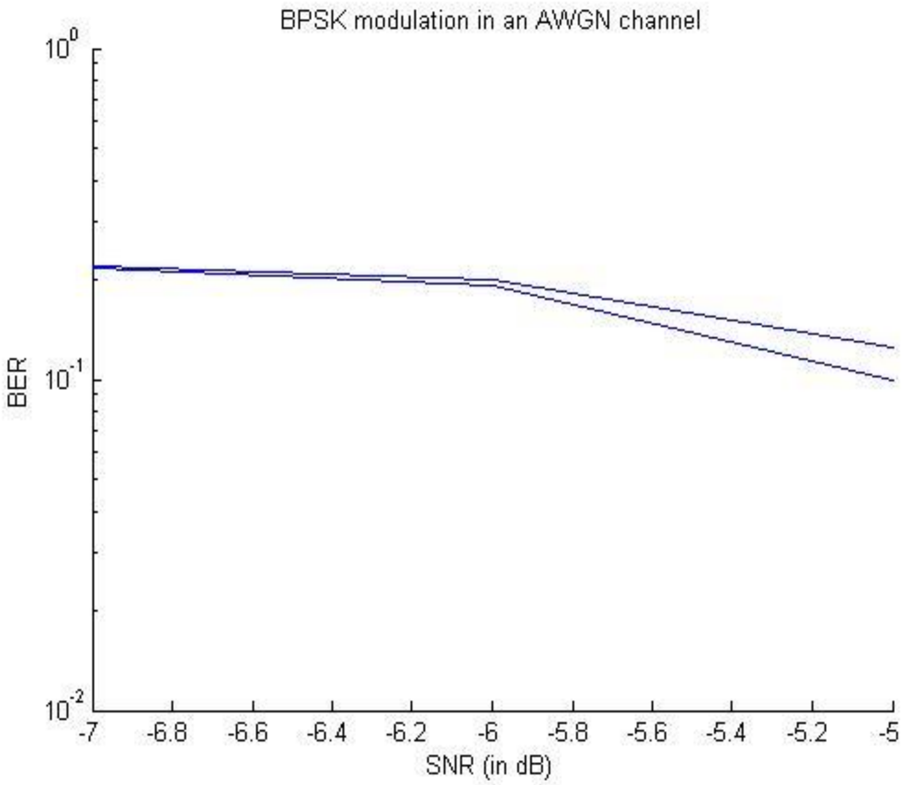
4.2.4 Transfer characteristics

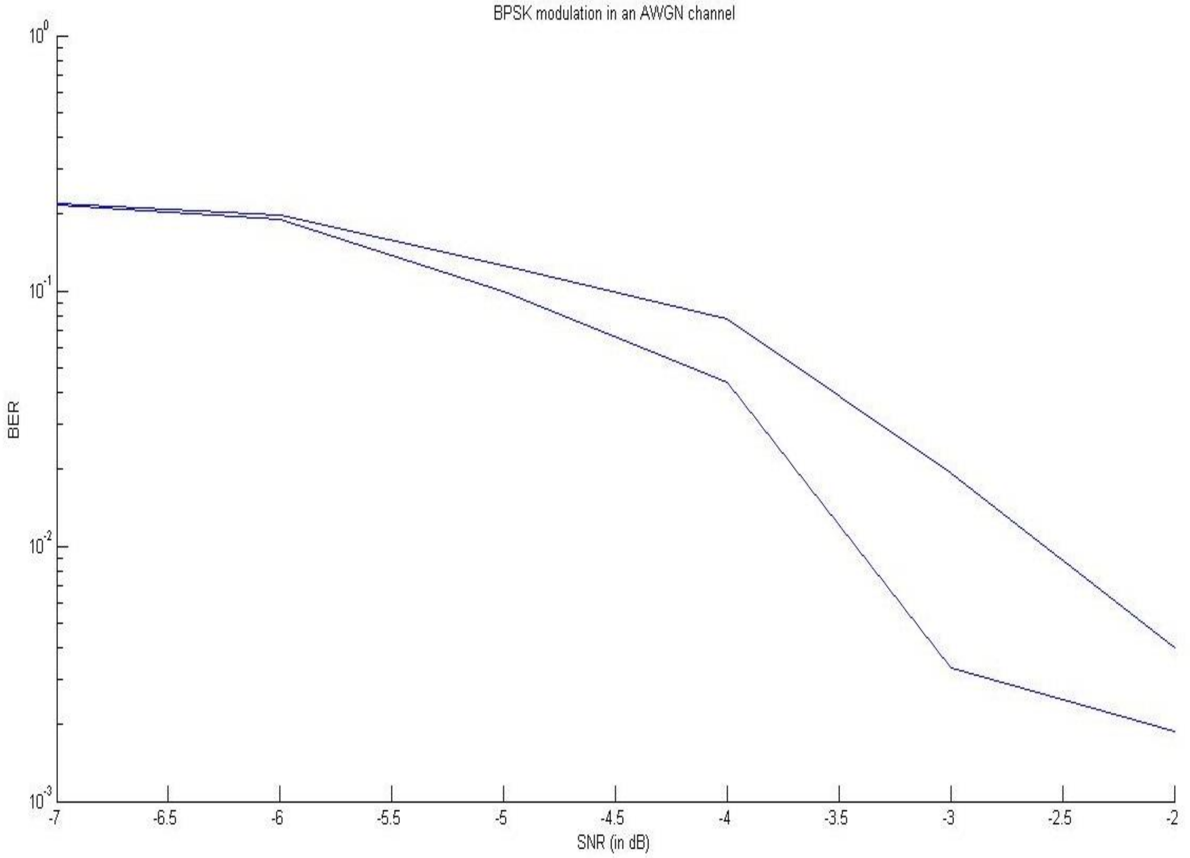
- Extrinsic and a priori information is traded by the iterative decoders. (E1 becomes A2 and E2 becomes A1)
- Mutual information help in tracing the messages ($I(L_E;X)$,$I(L_A;X)$)
- Many factors influence the transfer characteristics (for eg.no. of code memory of code, different polynomials of code etc.)

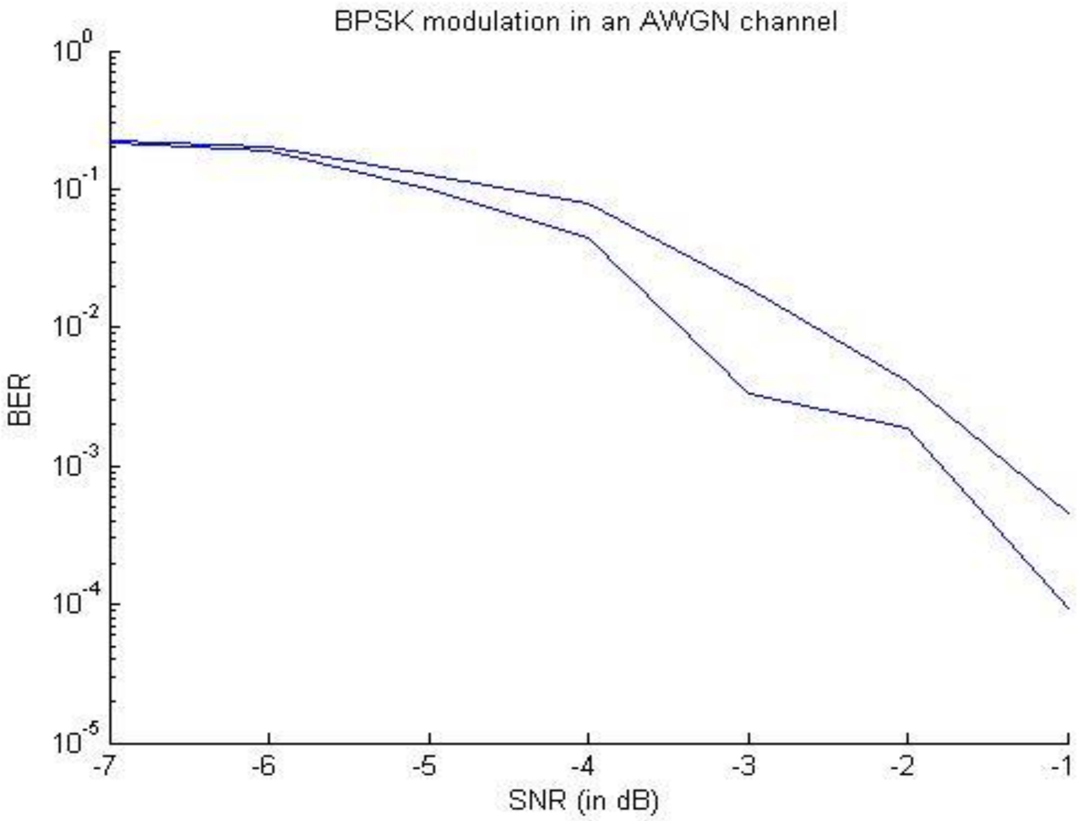
4.2.5 Simulation results

The simulation results compare the BER plots for turbo code and EXIT chart plots of turbo code.


BPSK modulation in an AWGN channel
BER
SNR (in dB)
$10^{-0.66}$
$10^{-0.67}$
$10^{-0.68}$
$10^{-0.69}$
$10^{-0.7}$
$10^{-0.71}$
-7 -6.9 -6.8 -6.7 -6.6 -6.5 -6.4 -6.3 -6.2 -6.1 -6

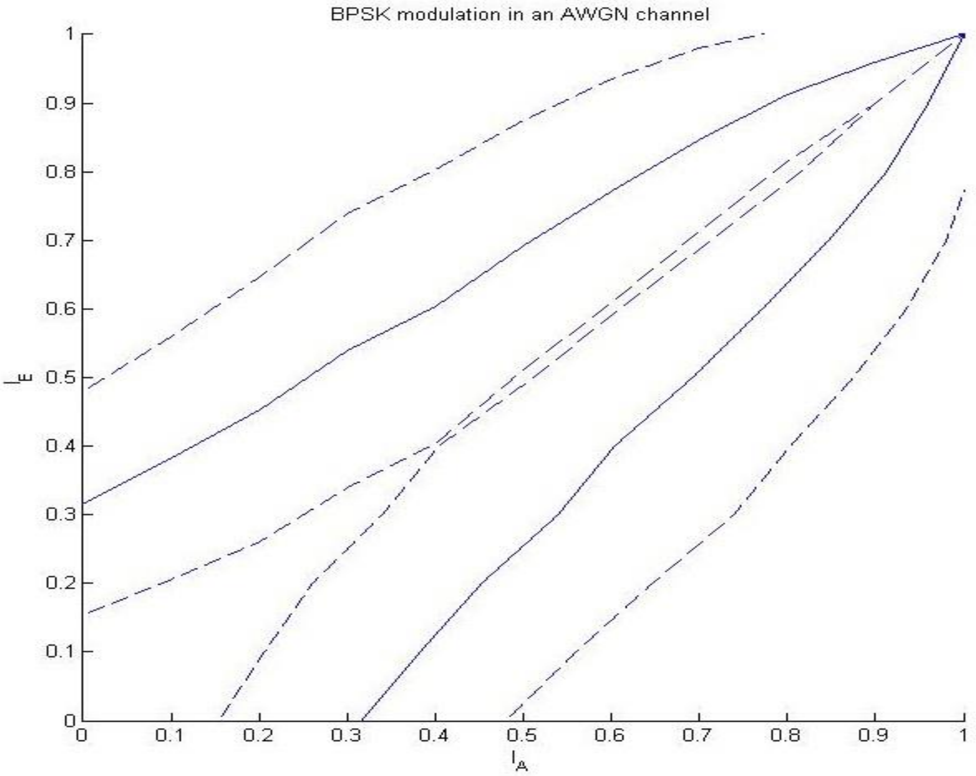

BPSK modulation in an AWGN channel
BER
SNR (in dB)
10^{0}
10^{-1}
10^{-2}
-7 -6.8 -6.6 -6.4 -6.2 -6 -5.8 -5.6 -5.4 -5.2 -5

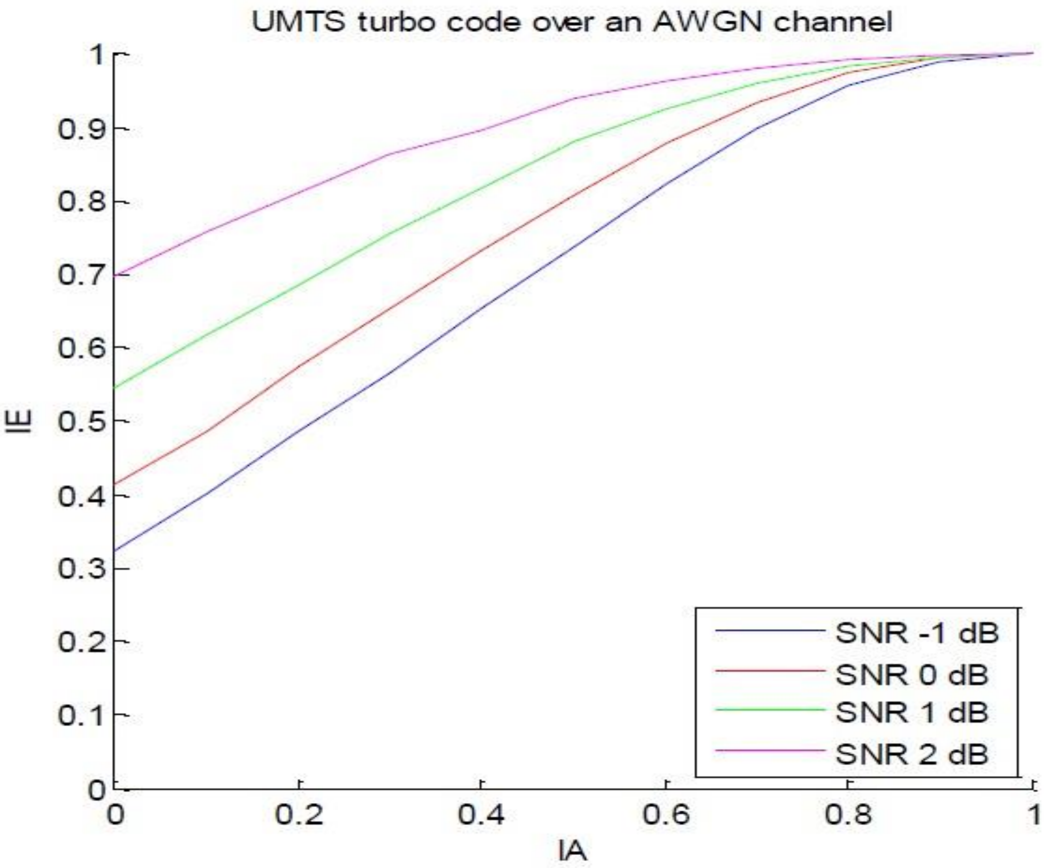




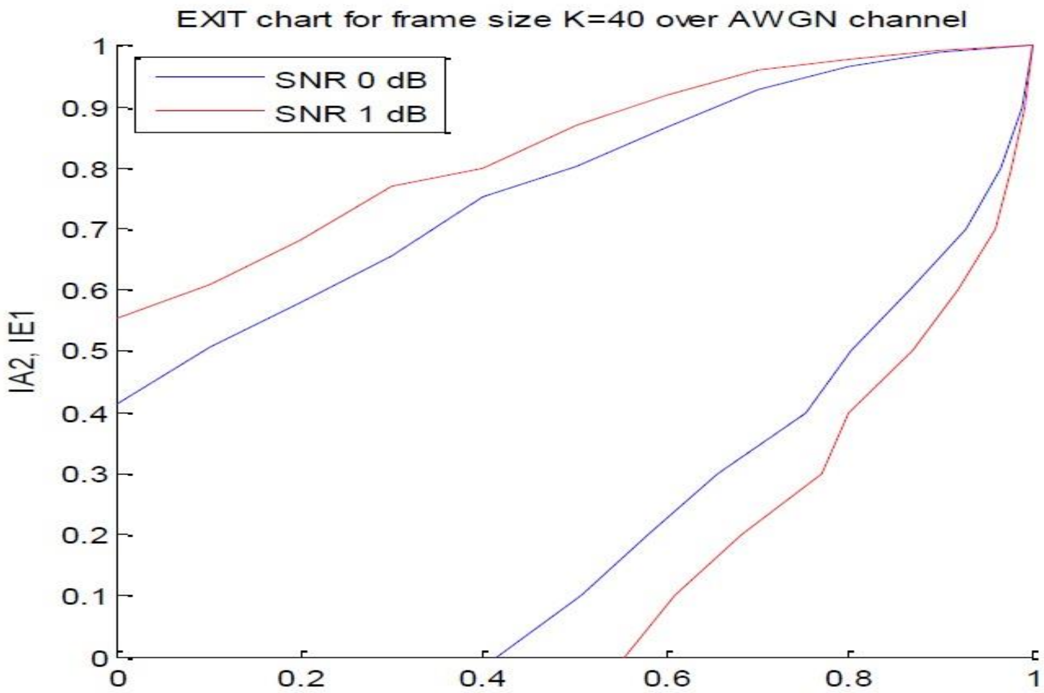
The above plots show the three different regions of BER plots i.e the Starting region(low values of SNR),second region(medium values of SNR) where the performance of the iterative decoding is very high and the third region(high values of SNR) where the decoding converges in few iterations. The plots show these properties for different ranges of SNR.

EXIT Chart plots:





Plot of EXIT charts for different SNRs



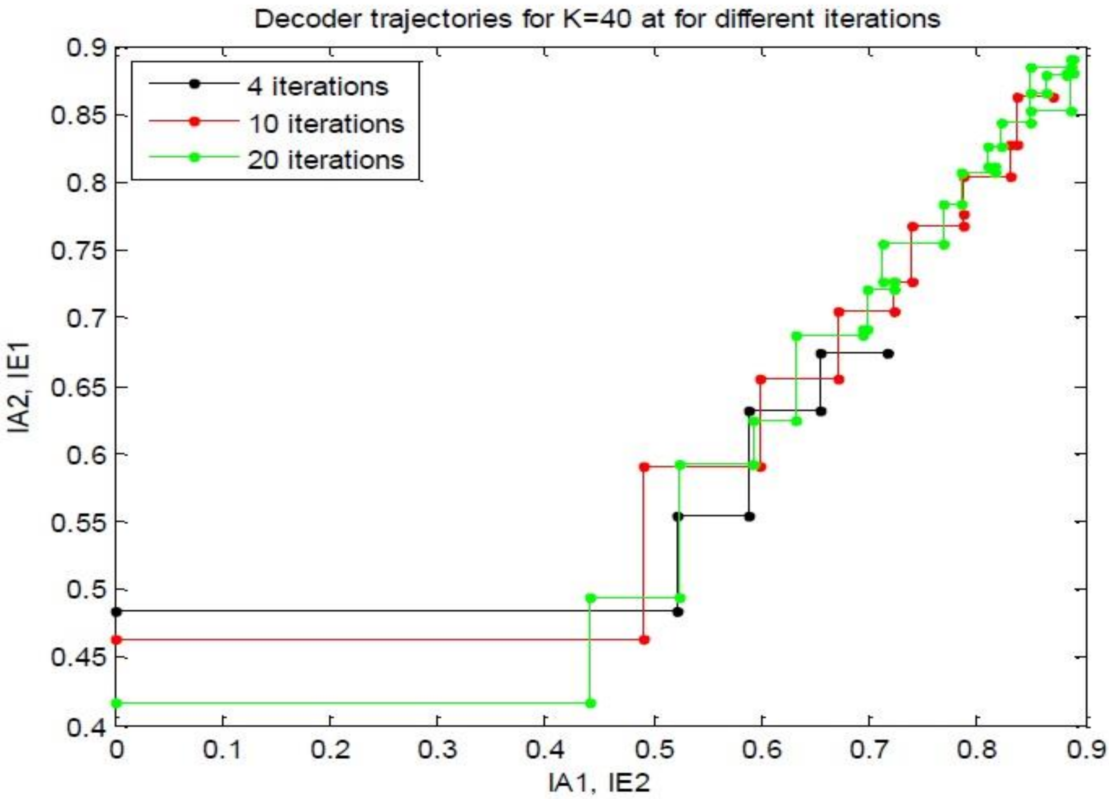
Plot of EXIT charts for different SNRs

4.3 DECODING TRAJECTORIES

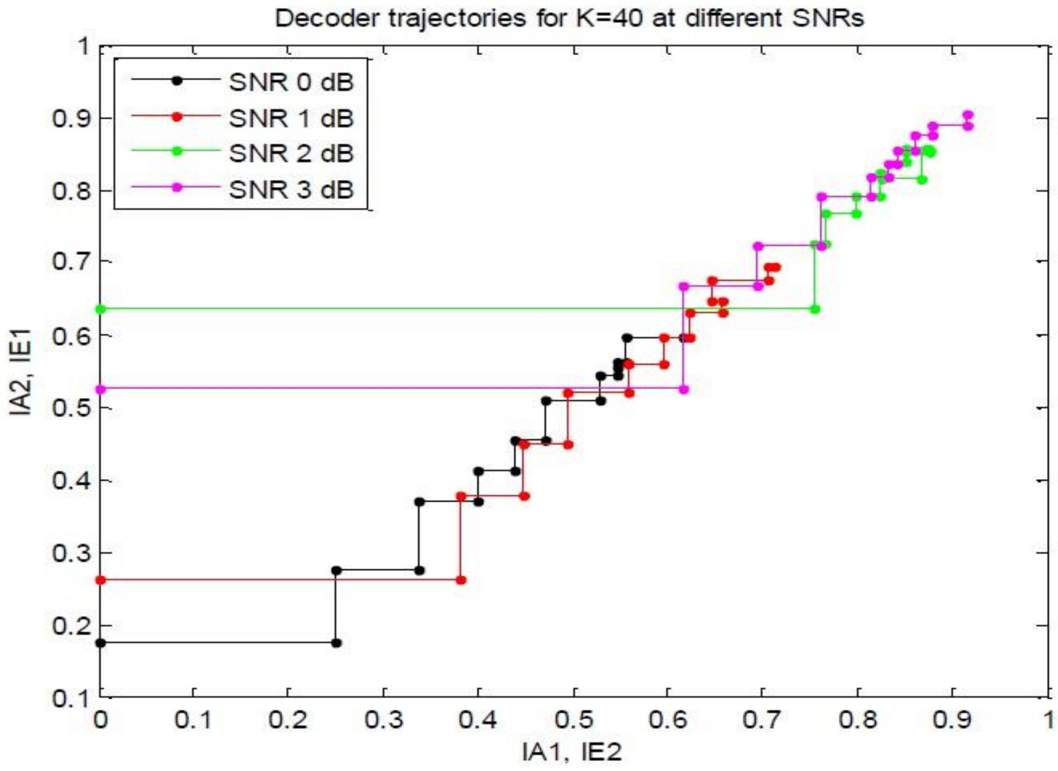
4.3.1 INTRODUCTION

- A single diagram shows the characteristics of both decoders. The second decoder's transfer characteristic is plotted by swapping the axes.
- The trading of extrinsic information is realised through decoding trajectories.

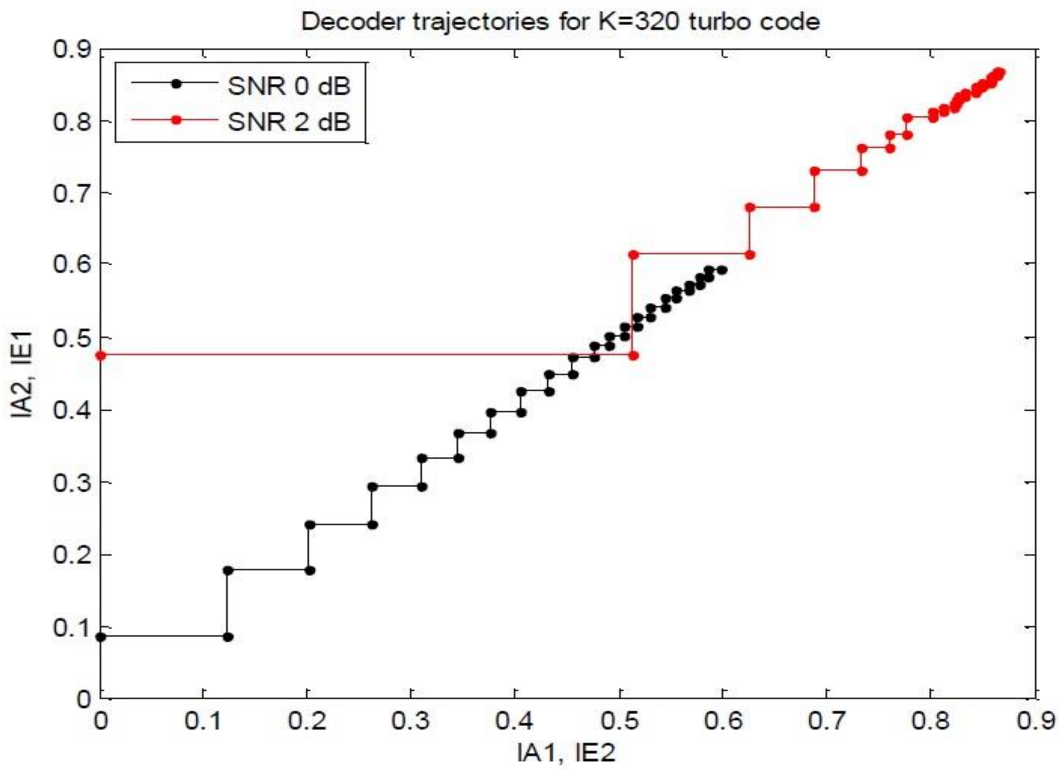
4.3.2 Simulation results



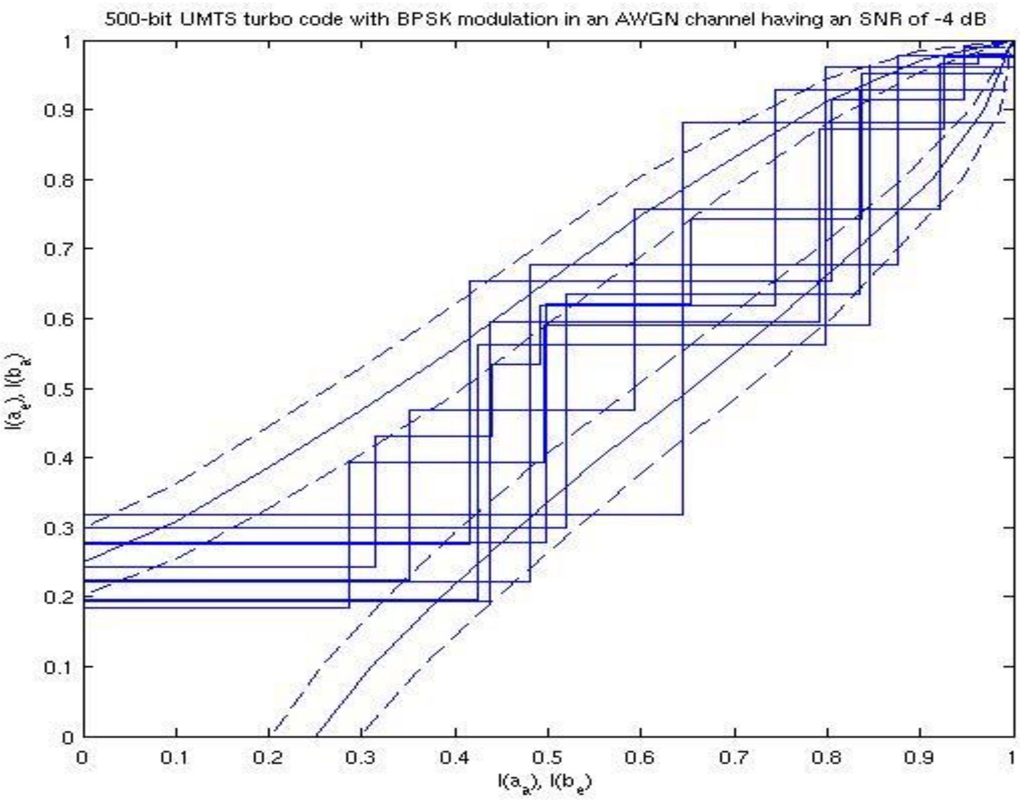
Decoding trajectory for different no. of iteratons-The trajectory reaches a farther point with increase in the no. of iterations.



Decoding trajectory for different values of SNR- As the SNR increases the a priori information of decoder 1 and the extrinsic information of decoder 2 increases.



Decoding trajectory for different frame sizes-With the value of SNR remaining constant,if we increase the frame size then the trajectories go nearer to the (1,1) point.



Combined EXIT Charts and decoding trajectories

4.3.3 CONCLUSION

This chapter analyses the performance of turbo codes by using a new technique which are EXIT charts. They provide much more additional information provided by the BER charts.

CHAPTER 5

REPEAT ACCUMULATE CODES

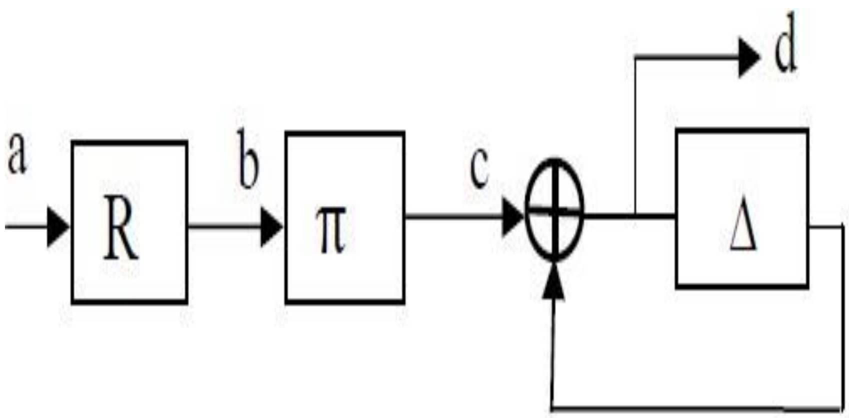
5.1 MOTIVATION AND INTRODUCTION

- Turbo codes consist of serial and parallel concatenated codes. RA codes is such a serial concateneated code.

- Every input bit is sent multiple no. of times to the encoder.

- RA codes are much simpler to operate than turbo codes.

5.2 BLOCK DIAGRAM OF THE ENCODER



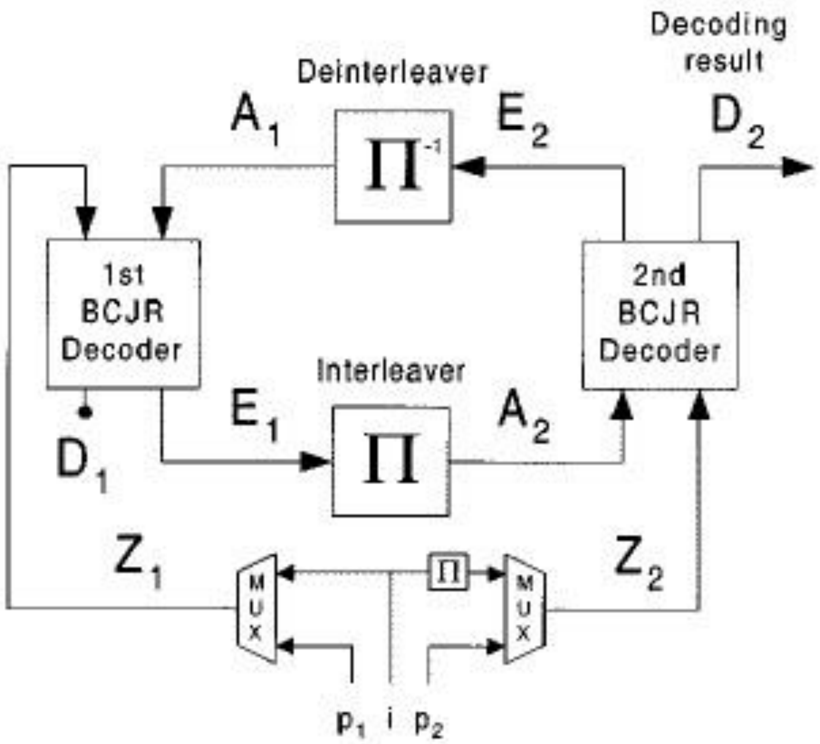
Here a is the input bit

b is the output of the repeater. It is given to the interleaver.

c and its delayed version are added(modulo-2) to produce the encoded bits ‘d’.

5.3 RA DECODER

To bring out a comparison between the turbo decoder and the RA decoder, the turbo decoder is shown again.



It consists of two BCJR decoders.

5.3.1 BLOCK DIAGRAM OF THE RA DECODER

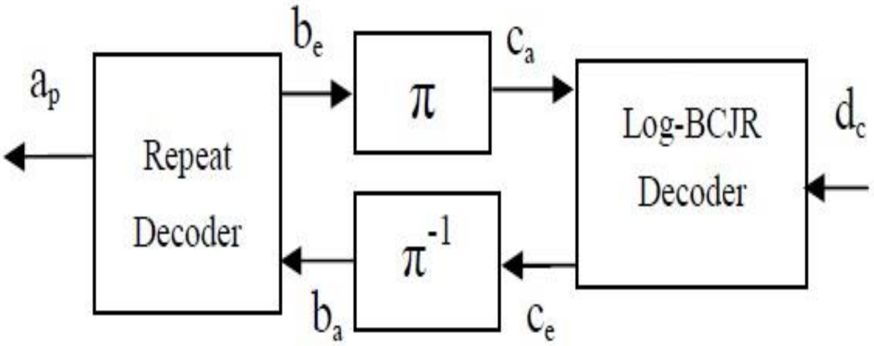


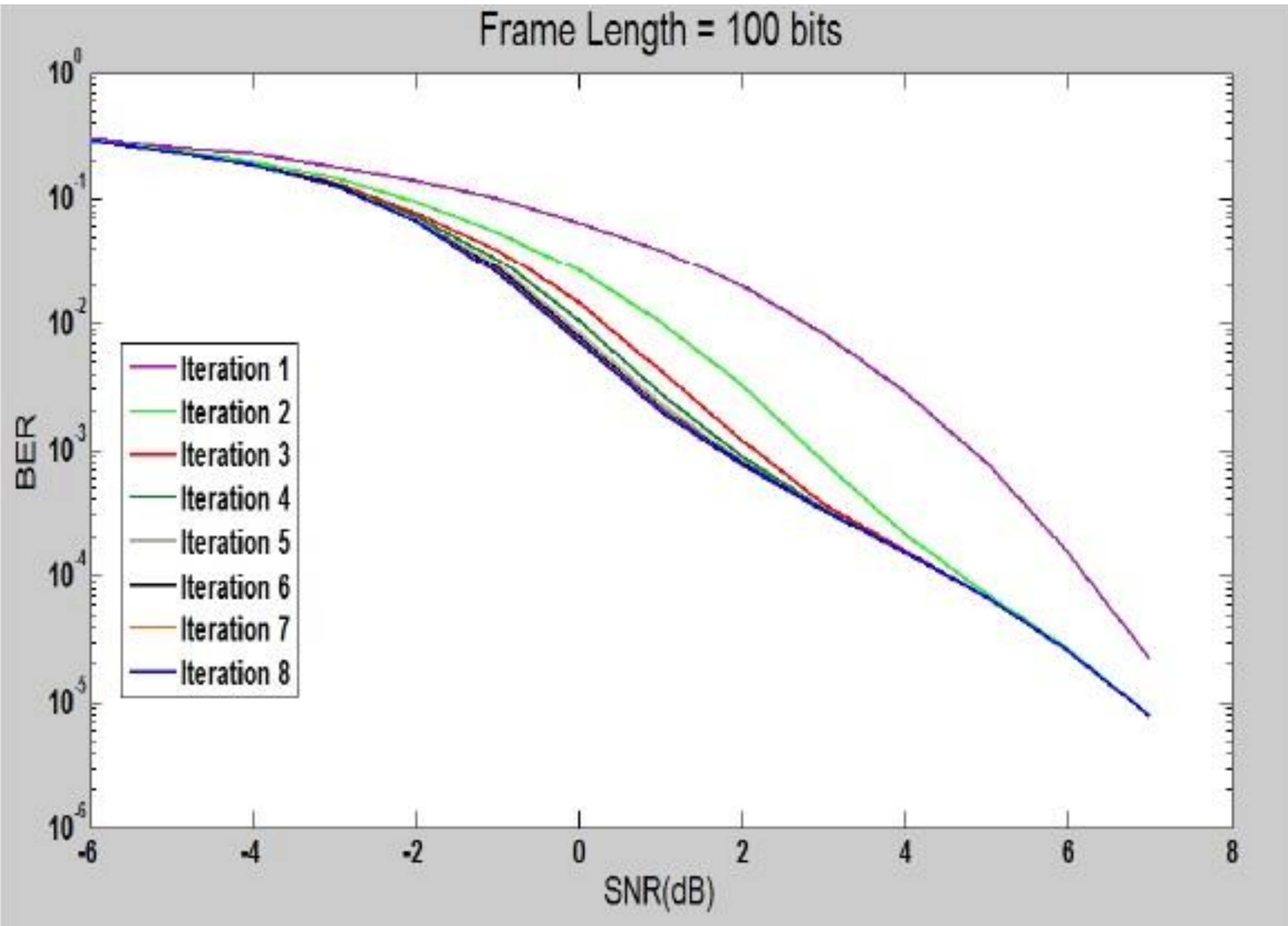
Fig 17:Block diagram of RA decoder

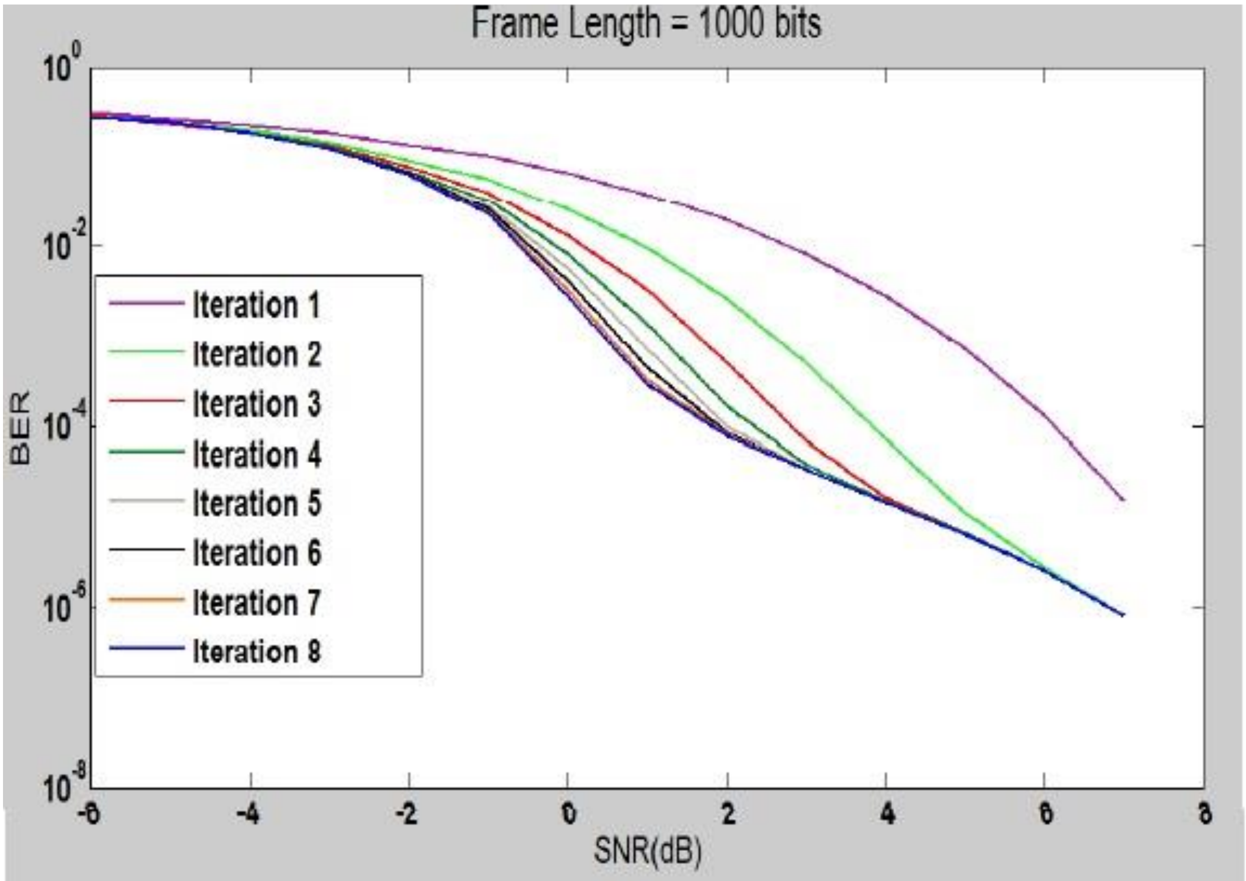
5.3.2 Procedure of decoding

- The decoder is fed by a priori LLRs (c_a) and the encoded input bits (d_c) to produce extrinsic LLRs (c_e), which are deinterleaved into a priori LLRs (b_a).
- To produce extrinsic LLRs (b_e), the repeat decoder swap each consecutive pairs of b_a.
- Before entering inside decoder, b_e is rearranged in the interleaver to compensate with the random interleaver, which is implemented in the encoder.
- The interleaved b_e is fed into the decoder as c_a. The decoder again is fed by two input bits c_a and d_c to compute c_e.
- To perform 1st iteration, c_a is initialised by setting it at zero value.
- A posteriori output, a_p is computed during the final iteration by adding each consecutive bits of b_a in pairs.

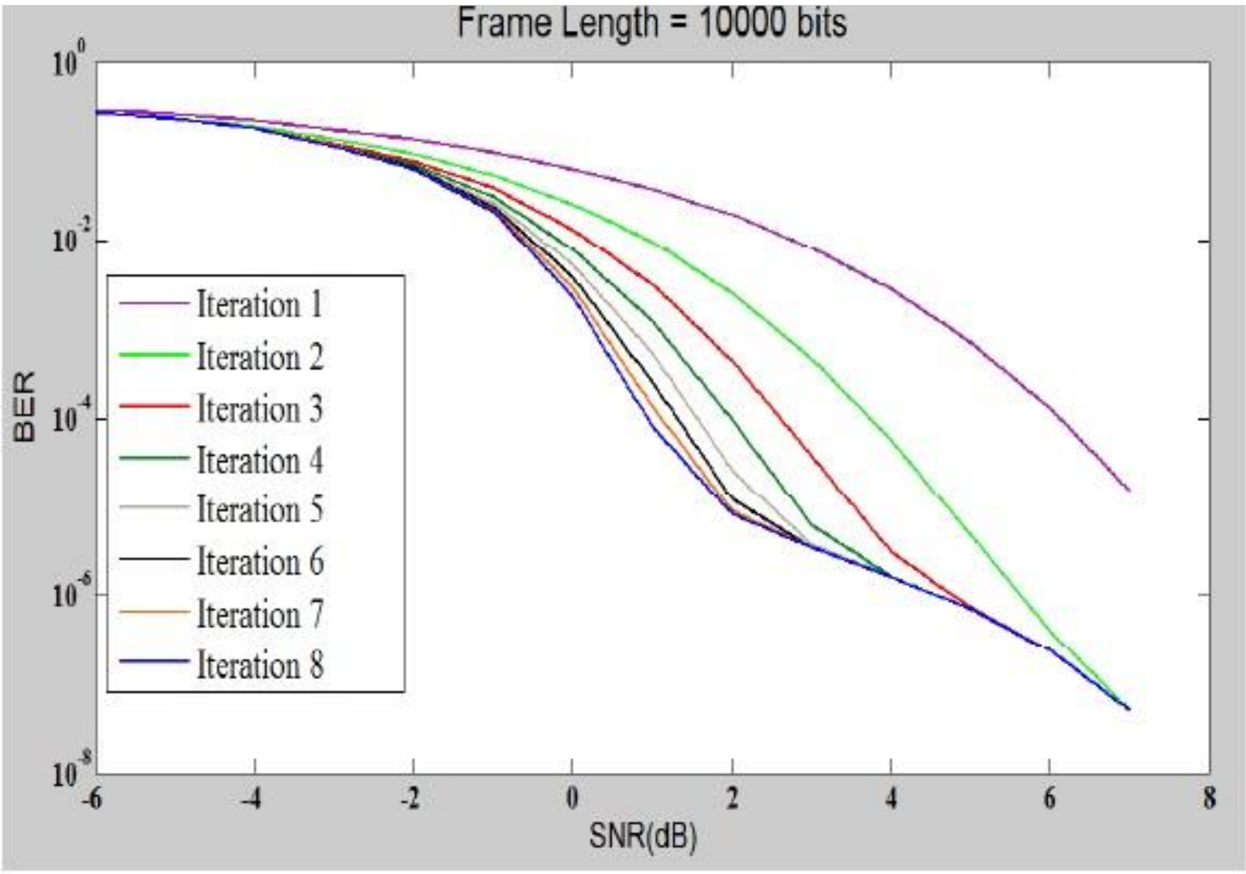
5.4 Simulation results

5.4.1 BER ANALYSIS





Frame Length = 1000 bits
BER
SNR(dB)
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8



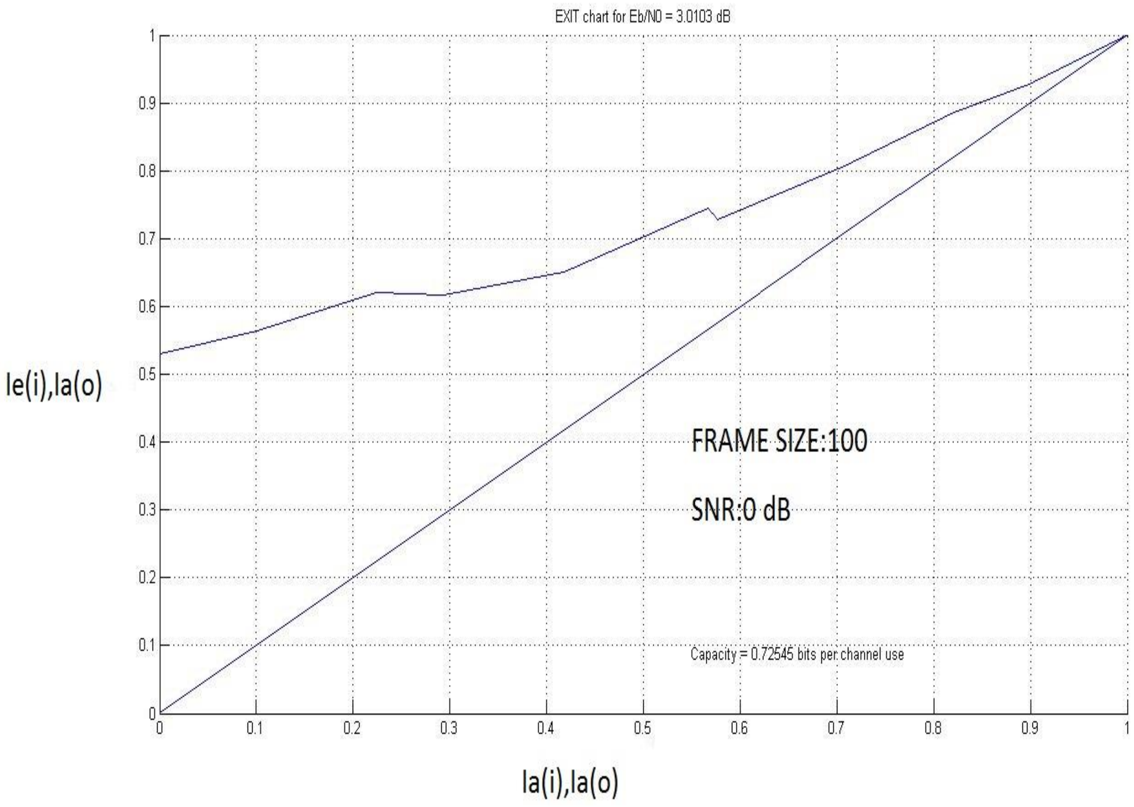
Frame Length = 10000 bits
BER
SNR(dB)
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8

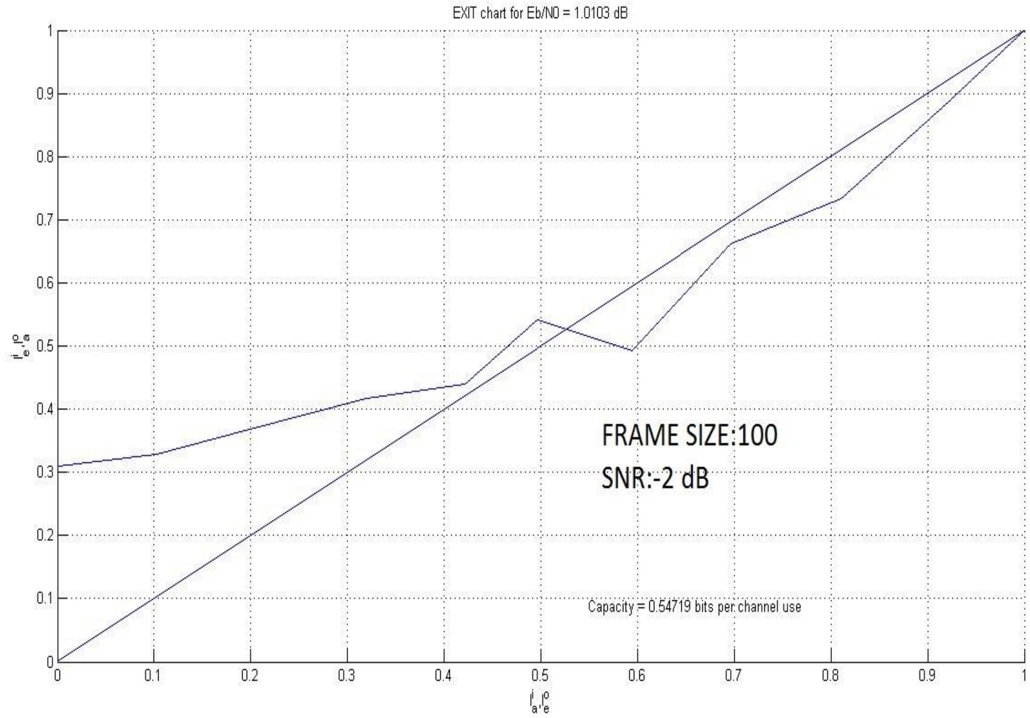
- As the no. of iterations increases the slope of the turbo cliff region increases.

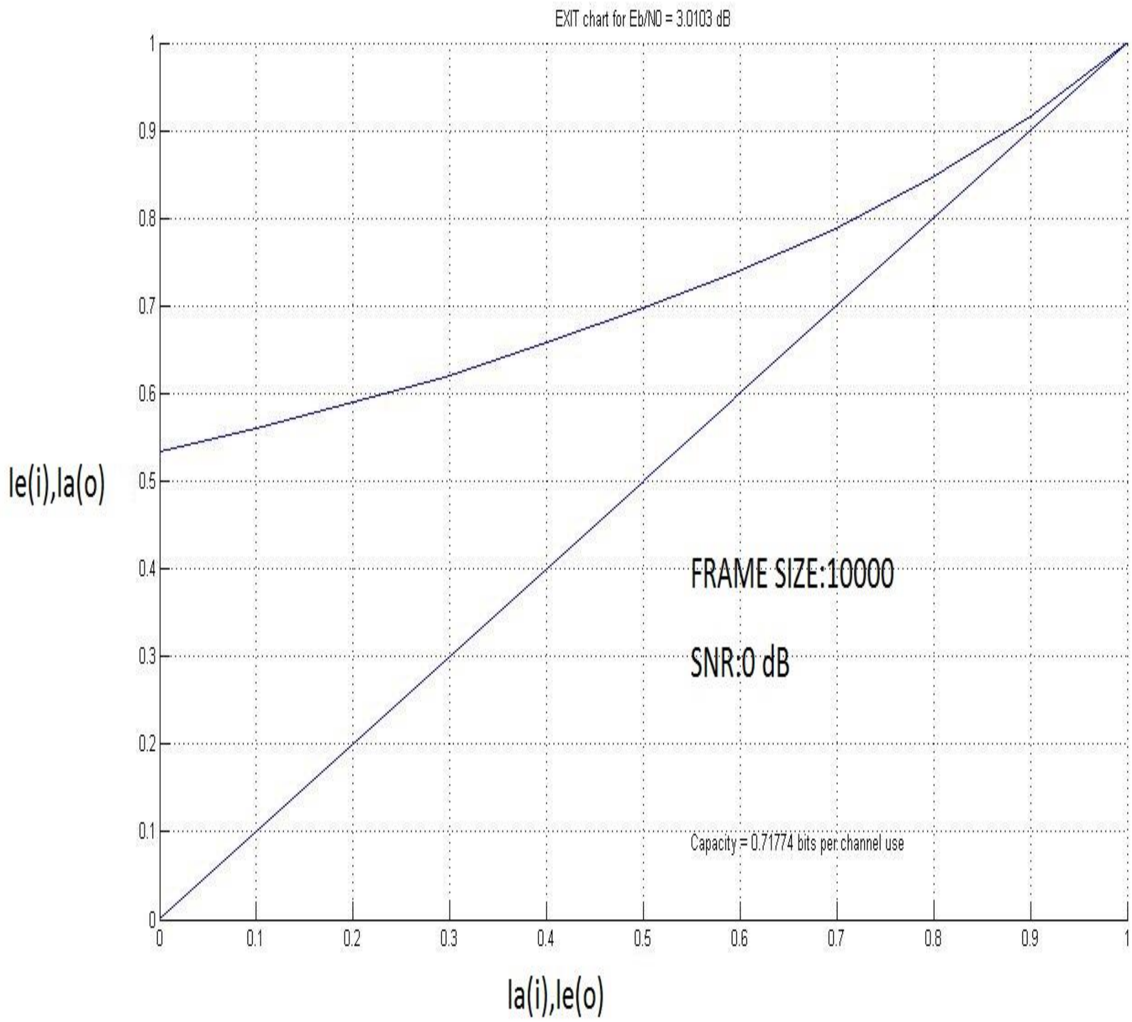
- The best BER performance is given by the eighth iteration when the frame length is 10,000 bits.

- The performance of the BER curve doesn't change much with an increase in the no. of iterations.

5.4.2 EXIT CHART ANALYSIS

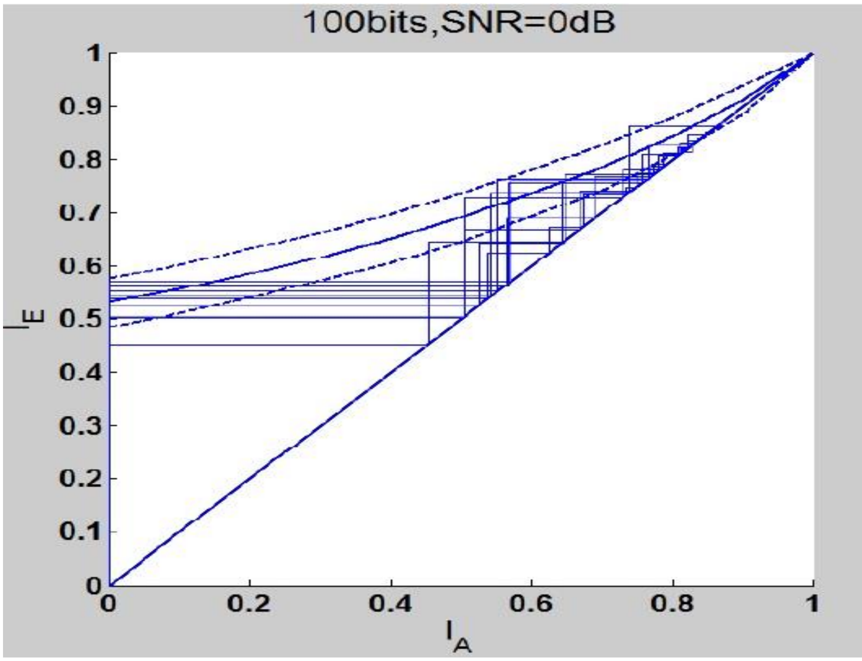


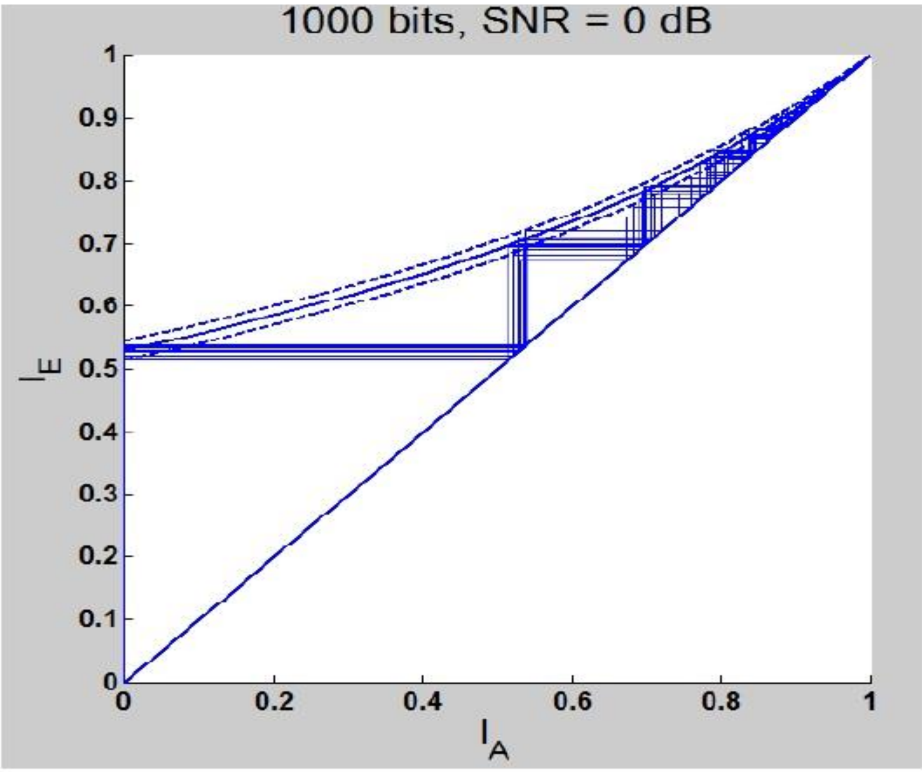

EXIT chart for Eb/N0 = 1.0103 dB
FRAME SIZE:100
SNR:-2 dB
Capacity = 0.54719 bits per channel use
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

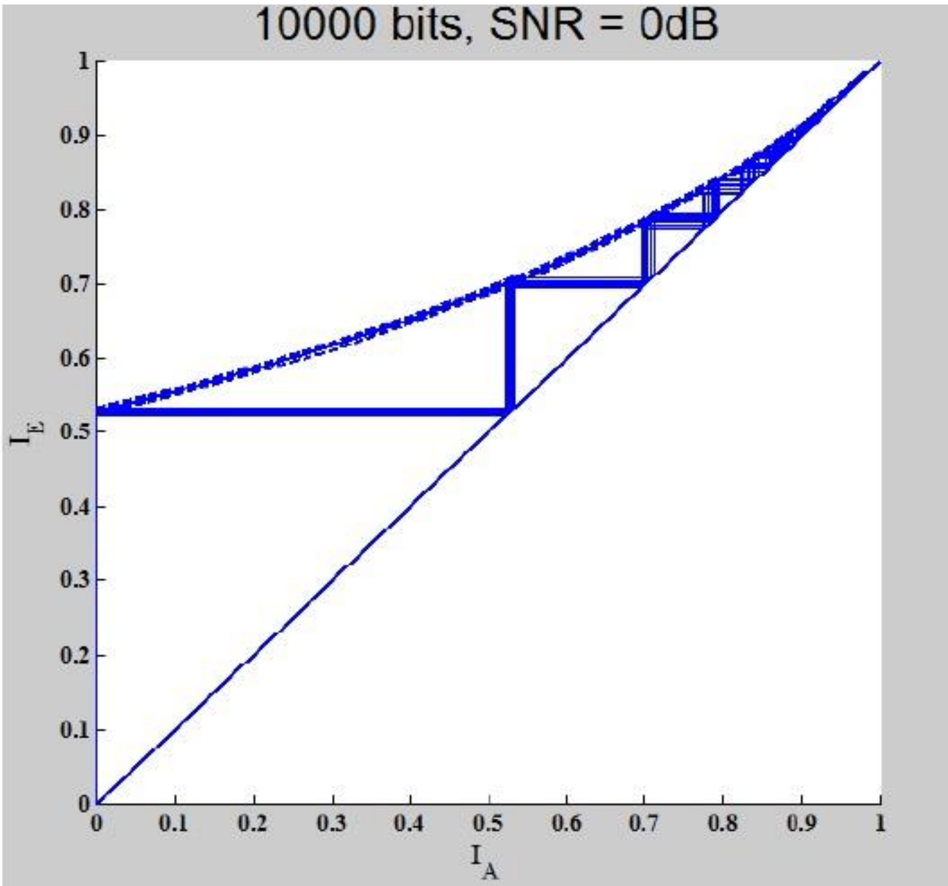


- The threshold SNR is not that prominent in finite length turbo like codes; as the BER curve has a wide smooth fall. The problem is solved by using the EXIT band chart.
- The variation of SNR controls the tunnel of the EXIT band,
- For 10000 and 1000 bits the trajectory sneaks through the tunnel to the (1,1) point and the iterative decoding estimates better output.

5.4.3 TRAJECTORIES







5.5 Conclusion

This chapter introduces a easier coding technique than the turbo codes called RA codes. This can be implemented with much less effort and complexity while provide nearly the same performance as turbo codes.

CONCLUSION

By examining the thesis we can find different methods of channel encoding and decoding methods.The methods discussed here are:

1.Convolutional coding and decoding 2.Turbo coding and decoding

3.Analysis of Turbo codes through EXIT Charts 4.Repeat Accumulate Codes

Convolutional codes showed better performance than the block codes. Turbo codes were an improvement of the Convolutional codes. The depth of the analysis of Turbo codes was increased through EXIT Charts.

Due to high degree of complexity of Turbo decoders, Repeat Accumulate codes were developed which showed nearly the same performance but decreasing the complexity hugely.

FUTURE WORK

VHDL Implementation of convolutional encoding and decoding

Optimizing of Iterative Turbo Equalizer for Underwater Sensor Communication

Efficient Parallel Turbo-Decoding for High-Throughput Wireless Systems.

EXIT charts of irregular codes

REFERENCES

G. Forney, “Maximum-likelihood estimation of digital sequences in the presence of intersymbol interference,” *IEEE Trans. Inf. Theory*.

L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inf. Theory*

J. Proakis, *Digital Communications*,

Mohammed El-Hajjar and Lajos Hanzo, “EXIT Charts for System Design and Analysis”, IEEE Communications Surveys & Tutorials, Vol. 16, no. 1, First quarter 2014

Stephan ten Brink, “Convergence behavior of iteratively decoded parallel concatenated codes”, IEEE Transactions on Communications, vol. 49, Oct. 2001.

Jorge Castiñeira Moreira,Patrick Guy Farrell,” ESSENTIALS OF ERROR-CONTROL CODING”, Copyright C @2006 ,John Wiley & Sons Ltd

http://users.ecs.soton.ac.uk/

Sahay Shamsey,” EXIT Chart Analysis of Repeat Accumulate Codes for Log-BCJR Algorithm in Iterative Decoding”, University of Southampton, ©2013 IEEE

Mohammed El-Hajjar and Lajos Hanzo, “EXIT Charts for System Design and Analysis”, IEEE Communications Surveys & Tutorials, Vol. 16, no. 1, First quarter 2014

Stephan ten Brink, “Convergence behavior of iteratively decoded parallel concatenated codes”, IEEE Transactions on Communications, vol. 49, Oct. 2001.